PROCEEDINGS

DMS 2005

The 11th International Conference on Distributed Multimedia Systems

Sponsored by

Knowledge Systems Institute, USA

Technical Program

September 5-7, 2005
Fairmont Banff Springs Hotel, Banff, Alberta, Canada

Organized by

Knowledge Systems Institute

ISBN 1-891706-17-9 (paper)

Additional Copies can be ordered from:
Knowledge Systems Institute Graduate School
3420 Main Street
Skokie, IL 60076
USA
Tel:+1-847-679-3135
Fax:+1-847-679-3166
Email:office@ksi.edu
http://www.ksi.edu

Printed in the United States of America

PROGRAM CO-CHAIRS' FOREWORD

As Program Co-Chairs, we would like to welcome you to DMS'2005 and we hope that you will have a productive and interesting time in the beautiful surroundings of Banff.

2005 marks the 11th annual Distributed Multimedia Systems conference. The initial DMS conference was held in Taipei in 1994, and the conference has been held in a number of countries in East Asia and North America since then. This year DMS is returning to Canada for the first time since 1997 when Vancouver hosted the conference. Since 1994, the field of Multimedia Computing has matured, with a greater coverage in academic course work as well as the introduction of a number of dedicated journals. DMS continues to publish work in both distributed and multimedia systems, especially the convergence of the two areas.

Over the years DMS has hosted a number of Workshops in areas related to the main Conference theme of distributed multimedia systems. This year is no exception, with the International Workshop on Visual Languages and Computing, the International Workshop on Mobile Systems, E-Commerce and Agent Technology, and the International Workshop on Distance Education Technology adding breadth to the topics which conference participants can explore.

A total of 141 papers were submitted. Of these, 45% were accepted as regular papers and 16% were accepted as short papers. Papers were submitted by authors in 14 different countries, with a well-balanced geographical distribution. 37 papers were by authors from Asia and Australia, 28 from Europe, and 20 from North America. DMS truly is an international conference.

We are truly grateful to the reviewers (who were primarily, but not solely, the DMS Program Committee members) whose diligent efforts have helped to increase the quality of the conference as well as of individual papers. Both the reviewers and the authors have worked under sometimes strict deadlines. We thank them for their timely work. We would also like to thank the organizers and Program Committees of the associated workshops as well as organizers of invited sessions and keynote speakers who have contributed greatly to the success of DMS'2005. We would also like to acknowledge the hard work that staff members at Knowledge Systems Institute, the conference organizer, have performed. Without their help, this conference would not have been possible.

A special thanks to Professor Shi-Kuo Chang whose inspiration and practical support have contributed so much to DMS this year, as they have since the initial DMS conference.

Angela Guercio
and Timothy Arndt
DMS'2005 Program Co-Chairs

The 11th International Conference on Distributed Multimedia Systems (DMS 2005)

September 5-7, 2005
Fairmont Banff Springs Hotel, Banff, Alberta, Canada

Organizers & Committee

Conference Chair

Shi-Kuo Chang, University of Pittsburgh, USA

Program Committee Co-Chairs

T. Arndt, Cleveland State University, USA
A. Guercio, Kent State University, USA

Program Committee

Arvind Bansal, Kent State University, USA
Athena Vakali, Aristotle University, Greece
Athula Ginige, University of Western Sydney, Australia
Augusto Celentano, University Ca' Foscari of Venice, Italy
Carsten Griwodz, University of Oslo, Norway
Chien-Tsai Liu, Taipei Medical College, Taiwan
David H. C. Du, Univ. of Minnesota, USA
Fadi P. Deek, New Jersey Institute of Technology, USA
Filomena Ferrucci, Univ. of Salerno, Italy
Fuhua Lin, Athabasca University, Canada
Genny Tortora, University of Salerno, Italy
Han-Chieh Chao, National Dong Hwa University, Taiwan
Ing-Ray Chen, Virginia Tech (VPI&SU), USA
James Kwok, Hong Kong University of Science and Technology, Hong Kong

Jean-Luc Dugelay, Institute EURECOM, France
Jonathan Liu, University of Florida, USA
Joseph E. Urban, Arizona State Univ., USA
Kai H. Chang, Auburn University, USA
Kang Zhang, The University of Texas at Dallas, USA
Larbi Esmahi, National Research Council of Canada, Canada
Makoto Takizawa, Tokyo Denki University, Japan
Mark Liao, Academia Sinica, Taiwan
Masahito Hirakawa, Shimane University, Japan
Maurizio Tucci, University of Salerno, Italy
Meng Chang Chen, Academia Sinica, Taiwan
Ming Ouhyoung, National Taiwan University, Taiwan
Mohamed Ally, Athabasca University, Canada
Nadia Berthouze, University of Aizu, Japan
Nikolay Mirenkov, The University of Aizu, Japan
Paolo Maresca, Univ. of Napoli, Italy
Peter Douglas Holt, Athabasca University, Canada
Rentaro Yoshioka, University of Aizu
Shih-Fu Chang, Columbia University, USA
Shi-Nine Yang, National Tsing Hua University, Taiwan
Shu-Ching Chen, Florida International University, USA
Son T. Vuong, Univ. of British Columbia, Canada
Steven L. Tanimoto, Univ. of Washington, USA
Stuart Goose, Siemens Corporate Research, USA
Syed M. Rahman, Minnesota State University, USA
Timothy K. Shih, Tamkang University, Taiwan
Vincent Oria, New Jersey Institute of Technology, USA
Wen-Syan Li, CCRL, NEC USA Inc.
William Grosky, University of Michigan - Dearborn, USA
Yonghee Choi, Seoul National Univ., Korea
Yoshitaka Shibata, Iwate Prefectural University, Japan
Zied Choukair, ENST Bretagne, France

Proceedings Cover Design

Gabriel Smith, Knowledge Systems Institute Graduate School, USA

Conference Secretariat

Judy Pan, Chair**,** Knowledge Systems Institute Graduate School, USA
Tony Gong, Knowledge Systems Institute Graduate School, USA
C. C. Huang, Knowledge Systems Institute Graduate School, USA
Rex Lee, Knowledge Systems Institute Graduate School, USA
Daniel Li, Knowledge Systems Institute Graduate School, USA

The 2005 International Workshop on Visual Languages and Computing (VLC 2005)

September 5-7, 2005
Fairmont Banff Springs Hotel, Banff, Alberta, Canada

Organizers & Committee

Program Co-Chairs

Philip Cox, Dalhousie University, USA
Trevor Smedley, Dalhousie University, USA

Program Committee

Shi Kuo Chang, University of Pittsburgh, USA
Anthony Cox, Dalhousie University, Canada
Jing Dong, The University of Texas at Dallas, USA
George Furnas, University of Michigan, USA
Zenon Kulpa, Institute of Fundamental Technological Research, Poland
Wei Lai, Swinburne University of Technology, Australia
Stefano Levialdi, Università di Roma, La Sapienza, Italy
Kim Marriott, Monash University, Australia
Rym Mili, The University of Texas at Dallas, USA
Joseph J. Pfeiffer, New Mexico State University, USA
Margaret-Anne Storey, University of Victoria, Canada
Genny Tortora, Università di Salerno, Italy
Guido Wirtz, Bamberg University, Germany
Kang Zhang, University of Texas at Dallas, USA

The 2005 International Workshop on Mobile Systems, E-commerce and Agent Technology (MSEAT 2005)

September 5-7, 2005
Fairmont Banff Springs Hotel, Banff, Alberta, Canada

Organizers & Committee

General Co-Chairs

Jason C. Hung, Northern Taiwan Institute of Science and Technology (KWIT), Taiwan
Gwo-Jong Yu, Aletheia University, Taiwan

Honorary Co-Chairs

Timothy K. Shih, Tamkang University, Taiwan

Program Committee Chair

Anthony Y. Chang, Oversea Chinese Inst. Of Tech., Taiwan

Publishing Chair

Chih-Yung Chang, Tamkang University, Taiwan

Advisory Committee

Pallapa Venkataram, Indian Institute of Science, India
Willy Susilo, University of Wollongong, Australia
Magdy Saeb, Arab Academy for Science, Technology & Maritime Transport, Egypt

Program Committee

Kamel ADI, Université du Québec en Outaouais, Canada
Junaid Ahsenali Chowdary, Center for Ubiquitous Systems, Korea
Tzung-Shi Chen, Chang Jung Christian University, Taiwan

Yuh-Shyan Chen, National Chung Cheng University, Taiwan
Lawrence Y. Deng, San John's and San Mary's Inst. Of Tech., Taiwan
Ramachandra A. Dixit, Satyam Computer Services Ltd. India
Hui-huang Hsu, Tamkang University, Taiwan
Sheng-Tun Li, National Cheng Kung University, Taiwan
Kuan-Cheng Lin, Northern Taiwan Institute of Science and Technology (KWIT), Taiwan
YingQun Liu, Tsinghua University, Beijing, China
Qusay H. Mahmoud, University of Guelph, Canada
Chikara Ohta, Kobe University, Japan
Chotirat Ratanamahatana, University of California at Riverside, USA
Prasan Kumar Sahoo, Van Nung Institute of Technology, Taiwan
Kuei-Ping Shih, Tamkang University, Taiwan
Ronggong Song, National Research Council, Canada
Jin Tamura, University of Tokyo, Japan
Gustavo A. S. Torrellas, Instituto Mexicano del Petroleo, Mexico
Kuniaki Uehara, Kobe University, Japan
Theodora Varvarigou, National Technical University of Athens, Greece
San-Yuan Wang, I-Shou University, Taiwan
Eric Wong, The Univ. of Texas at Dallas, USA
David Yang, National Kaohsiung Normal University, Taiwan
Chi Zhang, Florida International University, USA

Workshop Secretary

Ai-Chuan Wu, St. John's and St. Mary's Institute of Technology, Taiwan

The 2005 International Workshop on Distance Education Technologies (DET 2005)

September 5-7, 2005
Fairmont Banff Springs Hotel, Banff, Alberta, Canada

Organizers & Committee

Steering Committee

S. K. Chang, University of Pittsburgh, USA
Peter Tuen-Ho Yang, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Timothy S. Shih, Tamkang University, Taiwan, R.O.C.

Workshop Chair

Lawrence Y. Deng, St. John's and St. Mary's Institute of Technology, Taiwan

Award Co-Chairs

Chein-Shing Wang, St. John's and St. Mary's Institute of Technology, Taiwan
Chih-Yung Chen, St. John's and St. Mary's Institute of Technology, Taiwan
Chun-Liang Hsu, St. John's and St. Mary's Institute of Technology, Taiwan

Advisory Committee

Paolo Maresca, University of Naples, Italy
Stefano Levaildi, University of Rome, Italy
Mahbubur Rahman Syed, Minnesota State University, USA
Son T. Vuong, Univ. of British Columbia, Canada
Jian-Hua Ma, The University Josie, Japan
Qun Jin, The University Aizu, Japan

Publication Co-Chairs

Lin-Her Jeng, St. John's and St. Mary's Institute of Technology, Taiwan
Alexander S. Chen, St. John's and St. Mary's Institute of Technology, Taiwan

Program Committee

Rwei-Ching Chang, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Ko-Shung Chen, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Kune-Yao Chen, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Ruei-Xi Chen, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Chuan-Feng Chiu, Panasonic Taiwan Laboratories Co. Ltd.
Teh-Sheng Huang, ChungHwa Telecommunication Co., Ltd.
Jason C. Hung, Kuang Wu Institute of Technology, Taiwan, R.O.C.
Jong-Hann Jean, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Qun Jin, The University Aizu, Japan
Dong-Liang Lee, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Stefano Levaildi, University of Rome, Italy
Sheng-Tun Li, National Kaohsiung First Univ. of Tech. and Sci., Taiwan, R.O.C.
Chang-Ching Lin, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Shieh-Shing Lin, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Jian-Hua Ma, The University Josie, Japan
Paolo Maresca, University of Naples, Italy
Jen-Kuei Peng, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.
Mahbubur Rahman Syed, Minnesota State University, USA
Son Vuong, Univ. of British Columbia, Canada
Chun-Chia Wang, Kuang Wu Institute of Technology, Taiwan, R.O.C.
Ching-Sheng Wang, Aletheia University, Taiwan R.O.C.
Sheng-Yuan Yang, St. John's & St. Mary's Institute of Technology, Taiwan, R.O.C.

Workshop Secretary

Ai-Chuan Wu, St. John's and St. Mary's Institute of Technology, Taiwan

Table of Contents

Invited Session on E-learning

E-commerce, E-education and E-entertainment

Web Servers and Services

Mobile Networks, Mobile Computing and Mobile Agents

Multimedia HCI

Feature-Based Human Recognition and Tracking

OS Support for Distributed Multimedia Systems

Multimedia Communications and Network Architectures

VLC Papers

MSEAT Papers

Internet Application and E-Commerce

Agent Technology

Bluetooth and Wireless Network

DET Papers

Educational Technologies

Communication Technologies

Intelligent Technologies

Intelligent Decision Support for the Design of Distributed Multi-Media Systems

Guenther Ruhe
iCORE Professor and Industrial Research Chair in Software Engineering
University of Calgary
2500 University Drive NW
Calgary, Alberta, Canada T2N 1N4
e-mail: ruhe@ucalgary.ca

ABSTRACT:

The area of distributed multi-media systems is studied from a decision-making perspective. Making informed decisions can often be the key difference between failure and success. How to assign services to different customers? What media are most appropriate for which situation? How to prioritize between different network architectures?

The paradigm of intelligent decision support is used to bring rational and intuitive decision-making together. Hybrid intelligence is suggesting to make synergy between human and computational capabilities is an organized way. The advantage of the human intelligence based approach is that it is able to better handle soft and implicit objectives and constraints. The advantage of computer-based approach is exactly where the human based approach fails: to cover a large portion of the solutions space. This hybrid approach is studied as part of an evolutionary problem solving process that is able to accommodate both crisp and tacit knowledge.

Digital Inpainting

Timothy K. Shih
Department of Computer Science and Information Engineering
Tamkang University
151, Ying-Chuan Road
Tamsui, Taipei Hsien
Taiwan 25137, R.O.C.
e-mail: tshih@cs.tku.edu.tw
Web: http://www.mine.tku.edu.tw/chinese/teacher/tshih.htm
fax: + 886 2 26209749
phone: + 886 2 26215656 x2743, x2616

ABSTRACT

Digital inpainting uses spatial or frequency information to restore partially damaged/removed photos and artworks. Digital image inpainting is an interesting new research topic in multimedia computing and image processing since 2000. This talk will cover the most recent contributions in digital image inpainting and image completion, as well as concepts in video inpainting. In addition to a quick survey, the presentation will cover several algorithms. Most restoration algorithms consider a picture as a single layer. The talk will cover a new approach, which divides a Chinese painting into several layers. Each layer is inpainted separately. A layer fusion mechanism then finds the optimal inpaint among layers, which are restored layer-by-layer. We apply the algorithm on Chinese and western drawing. The result shows a high PSNR value as well as a high user satisfaction. The demonstration of our work is available at: http://www.mine.tku.edu.tw/demos/inpaint.

Proceedings

The 11th International Conference on Distributed Multimedia Systems (DMS 2005)

Co-Editors

T. Arndt, Cleveland State University, USA
A. Guercio, Hiram College, USA

Adaptive Background Evaluation for Foreground Detection with Gaussian Distribution, a Fast Approach

Gianluca Bailo, Paivi Ijas, Marco Raggio, Fabio Sguanci

Department of Biophysical and Electronic Engineering
University of Genova, Via Opera Pia 11 A,
16146 Genova, Italy
{bailo, raggio, sguanci}@dibe.unige.it paivi.ijas@tut.fi

Abstract

Adaptive background updating is one of the methods used to detect moving objects in video sequences. Many techniques have been presented in this field but there are few mentions about the usage of these methods in real- time applications. We concentrate in the speed of the algorithm and present a method that is fast enough to be used in video surveillance systems. We started from the ideas presented by Gaussian distribution for background generation. Instead of using actively all the pixels in the image we divide the pixels into active and inactive ones. Gaussian distributions are used to model the history of active pixels and to state whether they belong to background or foreground. According to the classification of the previous active pixel also the inactive pixels are classified as a part of the background or foreground. We also reduce the frame frequency and use only every n^{th} frame in the image sequence to construct adaptive background.
This article is organised as follows: In Chapter 1 some of the previous work and their results are introduced. In Chapter 2 we first describe the method used by Stauffer and Grimson [3] and then present our new ideas. The results are explained in Chapter 3 and finally a conclusion is given.

1. Introduction and Previous Work

Background subtraction is a common method used to detect moving regions in computer vision systems. It simply means the segmentation of the current scene to background (stationary scene) and foreground (the moving objects of interest in the scene). One of the simplest ways to implement background subtraction is to use a constant model for the background. This model is then used to distinguish the moving regions from the background. However, there are many problems with such a method. It can not deal with changes in illumination, objects brought to or removed from the background, shadows, repetitive motion such as the leaves of the trees etc. To solve at least some of these problems more intelligent methods to adapt the background model to the changing environment have been introduced.

In order to deal with changing illumination Ridder et al. [1] used Kalman filtering to update the background model. Kalman filters are used to find the most probable background pixels that are then used to update the model. The method does not allow new objects to be added to background nor their removal, for example. Friedman and Russel [2] have invented a background subtraction method for vehicle detection. In their approach the pixels in the image are classified to three colour distributions corresponding to road, vehicle and shadow colours. These distributions are updated using the EM-algorithm. However this method can only be used for a particular scene for vehicle detection.

Stauffer and Grimson [3] used some ideas presented by Friedman and Russel [2] and they model the history of each pixel using Gaussian distributions. These distributions are used to classify the pixels to foreground and background pixels and thus to update the background model. Their method gave some good results with repetitive motions and different lightning conditions. However it suffers from slow learning of the background at the beginning and is not very fast. KaewTraKulPong and Bowden [4] have further improved the method presented by Stauffer and Grimson by using different kinds of

update equations and also by adding shadow detection. Also Power and Schonees [5] have presented some new ideas for the updating equations and made some study about different parameters. Some promising results have been gained with these methods, however, each of them still have their drawbacks.
Toyama et. al [6] have presented a three-level system for background adaptation. The algorithm works in pixel, region and frame levels in order to solve as many problems as possible. This method can deal well with changing lightning conditions, repetitive motions and new objects brought to background. However the speed of the algorithm can be questioned since it uses information from three different levels.

2. Our Method

Our aim was to find method that could be used in real-time, multiple-camera video surveillance system to distinguish the moving regions in the image. We decided to start from the ideas presented by Stauffer and Grimson [3] as their method seems to be one of the most used ones. Our goal was to be able to improve the speed of this method. The original method of Stauffer and Grimson is presented in detail in chapter 2.1 and in chapter 2.2 we introduce our improvements.

2.1 Modelling pixel histories with Gaussian distributions

In their article Stauffer and Grimson [3] use the idea to model the history of each pixel in the image by K Gaussian probability density distributions. The history of a certain pixel $\{x_0, y_0\}$ is be defined as a time series

$$\{X_1, ..., X_t\} = \{I(x_0, y_0, i): 1 \leq i \leq t\} \quad (1)$$

where I refers to the image sequence and X_i is the intensity value of the pixel $\{x_0, y_0\}$ at time instant i. For colour images X_i is a vector, for grey level images it is a scalar.
The probability to observe a certain pixel value within the history values of the pixel is determined as

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} \bullet \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (2)$$

where K refers to the number of Gaussian distributions used. $\omega_{i,t}$ is the weight parameter that is used to describe which part of the data is described by the i^{th} Gaussian distribution. η is a Gaussian distribution that has two parameters: μ_t is the mean of the Gaussian distribution at time t and Σ_t is the covariance matrix at time instant t. Stauffer and Grimson use 3 to 5 distributions to describe the history of each pixel.
An on-line K-means approximation is then used to update the parameters of the distributions as new information is gained from new frames. For each frame every new pixel value is compared against the existing Gaussian distributions. A new pixel is said to match a distribution if it is within 2.5 standard deviations from the mean of the distribution. If a pixel matched with one of the weighted Gaussian distributions the mean and the variance of this distribution are updated using the following equations

$$\mu_t = (1 - \rho)\mu_{t-1} + \rho X_t \quad (6)$$

$$\sigma_t^2 = (1 - \rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T(X_t - \mu_t) \quad (7)$$

where $$\rho = \alpha\eta(X_t \mid \mu_k, \sigma_k) \quad (8)$$

α is the learning rate that is defined by the user and is used to define the learning speed of the background updating. The mean and the variance of an unmatched distribution are not updated. Also the weight parameters of all the distributions belonging to the certain pixel are updated as follows

$$\omega_{k,t} = (1 - \alpha)\omega_{k,t-1} + \alpha(M_{k,t}) \quad (9)$$

where $M_{k,t}$ is 1 for matched distribution and 0 for a distribution that was not matched. If the current pixel did not match with any of the K distributions the distribution with smallest weight associated to that particular pixel is replaced by a new distribution. The mean of this new distribution is set to the value of the current pixel the variance of the distribution is set large whereas the weight parameter is given a small value.
In order to define which of the K Gaussian distributions describing the history of a pixel result from background and which ones from the foreground the distributions for each pixel are ordered by a factor ω / σ. The distributions that describe the background are expected to have a large weight parameter and small variance. Thus the B first distributions are marked as background distributions. B is defined with the help of the background threshold that is defined by the user and simply means the minimum portion of the data that is considered to result from the background pixels. Thus

$$B = \operatorname{argmin}\left(\sum_{k=1}^{b} \omega_k > T\right) \quad (10)$$

If a pixel is matched with one of these B distributions it is marked as a background pixel. Otherwise the pixel is considered as a part of a moving object and thus it is marked as a foreground pixel.

2.2 Our approach

Instead of using colour images as Stauffer and Grimson did, we decided to convert the images to greyscale and use these images for the background estimation. The usage of grey level images makes the calculation of variance easier and the algorithm becomes somewhat simpler also in other parts. Anyhow this did not make the algorithm fast enough so some further improvements were needed.

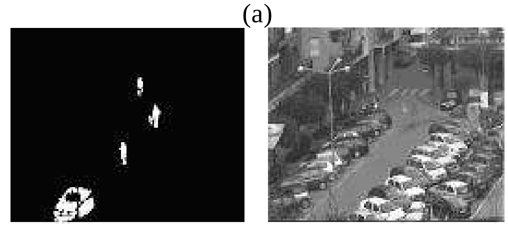
We left from the assumption that the values of neighbouring pixels have a correlation and thus also the history values of these pixels are correlated. Instead of modelling the history of all the pixels in the image we decided to use the information gained only from every other or every third pixel in the image. Thus the pixels are divided into active and inactive pixels. For example every other pixel in the image is used as active pixel and every other as an inactive pixel. The histories of active pixels are modelled with *K* Gaussian distributions as described in the previous chapter. Using these distributions it is then stated whether a new active pixel is a part of the background or the foreground. The inactive pixels are classified based again on the correlation assumption. If an active pixel was classified to be a part of the background also the inactive pixel/pixels following the active pixel are stated to belong to the same class. This way it is possible to reduce the number of Gaussian distributions needed to model the background which also reduces the elaboration time significantly.

We also decided to use only every n^{th} frame in the sequence to model the background. No crucial information is lost even if only every second, every fourth or every eighth frame is used for example. We ignored the frames between every n^{th} frame and used the interesting, every n^{th}, frames to create an adaptive background. At the end morphological operators were used to remove the unconnected pixels and to unit the connected ones. The result of the algorithm is a binary motion mask that shows the moving regions with white pixels and the background with black ones.

3. Results

Our system was tested with various video sequences containing repetitive motions, changes in lighting etc. The video sequences used included 25 frames per second. Three Gaussian distributions were used to model the history of each active pixel. All the tests were done using a 1,70GHz Pentium 4 with 256 MB RAM, which has Microsoft Windows 2000 as the operating system. Figure 1 shows the functionality of our system. Figure 1(a) presents the original scene with moving people and cars. Figures 1(b) and 1(c) show the binary motion mask when every pixel in the image is examined as in the original system presented by Stauffer and Grimson [3]. The results of using only every other pixel as an active pixel can be seen in Figures 1(d) and 1(e). Figures 1(f) and 1(g) show the same results when every third pixel is examined. As it can be seen in Figure 1 the binary motion mask barely changes even if the amount of active pixels is decreased. The shapes of the moving objects can be seen in the background a little more clearly when every other

(a)



(b) (c)

(d) (e)

(f) (g)

Figure 1*. Visual results. (a) The original image. Binary motion masks and background estimates*

when the information was collected from (b)&(c) every pixel, (d)&(e) every other pixel and (f)&(g) every third pixel.

or every third pixel is examined. However we are interested in a clear binary motion mask that can be used for motion tracking and can be calculated with a fast algorithm. Neither the changes made to the frame frequency do affect to the visual results. The binary motion mask is clear also when only every other, every fourth or every eighth frame of the sequence is used, for example. Next we studied the elaboration times needed per one frame using a different amount of active pixels and the results are shown in Table 1. As the results demonstrate the reduction in the amount of active pixels also significantly reduces the elaboration times. The variations in elaboration times are due to the different amount of movement in the scene. When using the information of every pixel 14- 16 frames per second can be elaborated. Instead of taking only every other pixel as active pixel 24- 26 frames per second and 30- 32 frames for every third pixel can be elaborated. The effects of our changes to the amount of noise in the binary motion mask were also studied.
The algorithm always creates some salt and pepper noise to the original binary motion image. This noise is due to the false detections made by the system and thus it is also a good measure for the robustness of the algorithm. The amount of noise created was approximated by calculating the difference image between the original binary motion image and the binary motion image after the usage of morphological operators.

	Only the adaptive backgroun d algorithm	*Adaptive background + morphological operations*
Every pixel active	40- 50 ms	50- 60 ms
Every other pixel active	20- 30 ms	30- 40 ms
Every third pixel active	10- 20 ms	20- 30ms

Table 1*. The times needed to elaborate one frame using our algorithm. Column 1 shows the times using only the algorithm for adaptive backgrounding and column 2 the times when also morphological operations to remove unconnected pixels and unite connected pixels are in use.*

The percentage of noisy pixels	*Every frame used*	*Every 2nd frame used*	*Every 4th frame used*
Every pixel active	1,60 %	1,61 %	1,63 %
Every other pixel active	1,62 %	1,61 %	1,62 %
Every third pixel active	1,60 %	1,62 %	1,63 %

Table 2*. The amount of noisy pixels in the image with different combinations.*

This difference image shows the false detections made by the system that are the pixels considered to be noise. The number of falsely detected pixels was calculated and compared with the total number of the pixels in the image. Table 2 shows the percentage of noisy pixels when a different number of active pixels and different frame frequencies were used. As it can be seen the amount of active pixels does not affect to the amount of noise created by the system. Neither does the usage of only every n^{th} frame. As the results show we succeeded in our aim to reduce the elaboration time of the algorithm. The improvements made do not affect either to the robustness or the visual results of the system. Some other problems of the algorithm presented by Stauffer and Grimson [3] still remain unsolved. One of the drawbacks is the time needed to learn the background at the beginning and also after sudden changes in lightning. Another problem is that when the moving objects stop they are adapted to the background too fast. These problems will be further studied and remain as future work.

4. Conclusions

We presented an adaptive background method based on the model presented by Stauffer and Grimson [3]. In our method only every other or every third pixel in the image was modelled by Gaussian distributions. The results show that our method is faster than the original system. The ability to detect moving areas and the amount of noise in the binary motion image do not differ from the properties of the original system. Due to the speed and robustness of our method it can be implemented to a real life video surveillance application and can be run with normal computers. We concentrated in the speed of the algorithm so some other problems such as slow learning at the beginning or too fast adaptation of still objects still remain to be solved efficiently in the future.

References

[1] Ridder, C., Munkelt, O., Kirchner, H. 1995. Adaptive Background Estimation and Foreground Detection using Kalman- Filtering. *Proceedings of International Conference on recent Advances in Mechatronics*, ICRAM'95. June 1995. UNESCO Chair on Mechatronics. Pages 193- 199.

[2] Friedman, N., Russel, S. 1997. Image segmentation in video sequences: A probabilistic approach. *In Proc. of the Thirteenth Conference of Uncertainty in Artificial Intelligence (UAI)*. August 1- 3 1997.

[3] Stauffer, C., Grimson, W.E.L. 1999. Adaptive background mixture models for real- time tracking. *In IEEE Conference on Computer Vision & Pattern Recognition*. Colorado, USA. June 1999. IEEE. Pages 246 – 252.

[4] KaewTraKulPong, P., Bowden, R. 2001. An Improved Adaptive Background Mixture Model for RealtimeTracking with Shadow Detection. In Proc. of 2nd European *Workshop on Advanced Video Based Surveillance Systems, AVBS01*. Kingston upon Thames. Sept 2001.

[5] Power, P.W., Schoonees, J.A. 2002. Understanding Background Mixture Models for Foreground Segmentation. *Proceedings Image and Vision Computing. New Zealand*.

[6] Toyama, K., Krumm, J., Brumitt, B., Meyers, B. 1999. Wallflower: Principles and Practice of Background Maintenance. *International Conference on Computer Vision*. Corfu, Greece. September 1999.

Composite Algorithms in Image Content Searches

J. R. Parker
Digital Media Laboratory
Department of Computer Science
University of Calgary
+1 403 220 6784
parker@cpsc.ucalgary.ca

Brad Behm
Computer Science Department
University of Calgary
2500 University Dr. N.W.
Calgary,Alberta,Canada T2N 1N4
1 403 220 6784

1. ABSTRACT

A combination of three algorithms is proposed that gives a reasonable rate of success in image similarity searches. All use region based statistical measures. They are combined using a simple decision fusion voting scheme, giving success rates in the area of 50-60%. For a web search application this is quite reasonable.

2. INTRODUCTION

It has long been a goal of research to identify images in a data collection that are similar to one given as an example. "Find me more images like this one" is the question, and it is an important question to answer. In spite of this, most similarity searches actually available to the users of search engines are based on text searches of captions, or use manually classified images. To be able to conduct a web search for images 'that look like this one' is especially desired by students and researchers in the humanities and social sciences, who generally have less ability to draw on other technical and computational resources than do the scientists and engineers.

During the summer of 2003 and in the time since, the Digital Media Laboratory at the University of Calgary became involved in a research project concerning to the creation of an educational object repository. These objects range in complexity from simple text files and word processor documents to complex interactive multimedia systems, animations, and games; The ability to search this repository for specific objects of interest was a key feature, especially as the repository grows in size and complexity. Since our laboratory has extensive experience with images in general, and computer vision in particular, it was thought that we could create a prototype image search engine that could be used in a practical context to access objects across the internet.

How does one know if an image, really just a collection set of RGB pixels, is similar to another image? General object recognition is not yet feasible, and it seems that this may be the way that we humans handle the problem. We need a practical way to compare images in an objective manner, quickly, and with a reasonable rate of success. Of course, it is obvious that two different images even of the same scene will never actually be identical. The random 'noise, component of each would be unlikely to agree, if nothing else were different, but illumination, geometry, orientation, and occlusion all play a part in complicating the issue. A complete solution to the problem is intractable.

Our approach was one that we have used in other situations, and which has served very well in producing high success rate prototypes in the past. We use multiple algorithms, collections of methods that we know have potential to work in this context, and which have been evaluated at some time in the recent past by more than one researcher. We use all of these methods in a decision fusion scheme, first having each algorithm arrive at a decision and then joining all the decisions into a single one. This approach has worked in the past for handprinted symbol recognition and signature verification, among other tasks. If the individual methods are simple and quick, then the overall method will be also, and the if the individual success rates are high enough then so will be the success rate of the composite system, higher than any of the individual success rates of the component algorithms.

3. PREVIOUS WORK

We have not found any previous work in this precise area - the use of multiple statistical algorithms in the context of image similarity. There are, of course, a large collection of individual similarity algorithms published, especially in the past six or seven years.

Del Bimbo [1] provides what is likely the best single source of information in this area, describing dozens of algorithms of various types and evaluating and comparing them. He groups methods into four basic types: those using color similarity, those using shape similarity, texture, and spatial relationships. Texture similarity is much like color similarity in principle, and is usually computationally expensive, so this method will not be pursued at length. Likewise, methods using spatial relationships will probably be relatively slow, and will take a significant time to implement correctly.

3.1. Color

Color is a practical and effective image property that can be used for similarity searches [10,12], and there are many ways to use color. We have been interested in histogram based techniques for other applications (E.G. [8]), and were therefore intrigued by the use of color histograms described in multiple sources [3,5,13].The basic idea is to use a quantization scheme to reduce the number of colors in the image, then to count how often each color occurs (number of pixels of that color). This is a simple characterization of the image that can be compared with others.

Of course, some images do not have color, and so we may have to apply these histogram methods to simple grey levels, counting the number of pixels having each grey value. In most cases color images can be converted into grey scale without the loss of region or shape information. Black and white movies and television were, after all, the state of the art many years ago, and it was always possible to see clearly what was happening - scenes and objects can be distinguished without using color.

4. A MULTIPLE ALGORITHM SYSTEM

We will design a composite system, one that uses multiple algorithms, that uses a selection of color based and simple shape feature based techniques. Given that the educational object repository project gave us a limited time frame for experimentation and implementation, it was decided to use methods that did not require a segmentation step.

Not only are we using multiple algorithms, but we apply them to sub-regions of the images, and so we tested multiple schemes for defining these regions. We have selected five algorithms for the implementation, although other methods were tried and discarded for various reasons. For each algorithm we implemented five methods of defining regions on the image, and compared these against each other using each similarity algorithm.

Each similarity algorithm uses a simple image property or measure. We compute these values for all of the images in our collection, then classify incoming targets using a nearest neighbor method; the image in the database having a measure m_i most like that of the query image I_i is said to be most similar to I.

The five measures we used were: grey sigma, edge density, boolean edge density, edge direction, and color histograms, and are described below.

Grey Sigma - Variation in Intensity

This has been described as a simple texture metric, but really is just a measure of the variability of the brightness as captured in the pixel values. The intensity variation across a region is determined by calculating the standard deviation of the intensity values of all of the pixels. If the image is color, then the pixels are converted into grey values using a standard HSI conversion [1,6] or by simply averaging the R,G, and B values.

Edge Density - Boundaries Between Objects

This is a simple geometry measure based on the strength of the edges in a small image region. It can be found by first using a standard edge detector (E.G. Sobel[6]) to enhance the pixels that belong to edges and boundaries. The result is a set of pixels whose values represent the strength of the edge at that point. Pixels far from an edge are 0, those near and edge increase to a maximum value.

The edge density measure is calculated as the mean pixel value of the edge enhanced image.

Boolean Edge Density

This is, on the face of it, very similar to the edge density method above. After the edge detector has been applied to the image, the image is thresholded so that what could be called edge pixels are white (1) and non-edge pixels are black. The measure returns the proportion of white (edge) pixels in the region. The difference between this and standard edge density is seen in images with noise. Boolean edge density sometimes allows noise to be minimized; edges are also high frequency information, and the thresholding process in Boolean edge density tends to reduce the effect of noise.

Edge Direction

Many edge detectors, including the Sobel edge detector, can be implemented as a convolution of a small (E.G. 3x3) image mask, often multiple masks each with a directional bias. This allows a crude estimate of edge direction to be made. In particular, for a typical 3x3 region in an image the two Sobel masks are:

$$S_x = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \qquad S_y = \begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix}$$

Given that the response to each mask represents a vector in the X or Y direction, the direction associated with the pixels in the region can be established using simple trigonometry:

$$\theta = \arctan (S_y/S_x)$$

This edge direction value is used to compute an overall estimate of the direction of the edges in a region by calculating a resultant vector over all pixels in the region.

When similarity between images is calculated it is done using differences between these region-based resultants.

Hue and Intensity Histograms

The technique described in [13] for creating color histograms has the rather desirable property that it disregards achromatic information, often included as noise in other types of color histogram. This is accomplished by calculating the standard deviation σ of the red, green, and blue components of a color pixel and then normalizing to the range [0,1].

The *chrominance* of a pixel is found using the function

$$\mu(\sigma) = \begin{cases} 0 & \text{if } 0 \le \sigma < a \\ 2\left(\frac{\sigma - a}{b - a}\right)^2 & \text{if } a \le \sigma < \frac{a+b}{2} \\ 1 - 2\left(\frac{\sigma - a}{b - a}\right)^2 & \text{if } \frac{a+b}{2} \le \sigma < b \\ 1 & \text{if } b \le \sigma < 1 \end{cases}$$

where a and b are constants between 0 and 1, where a<b. In the experiments that we conducted, we found useful values to be a=0.05 and b = 0.8. These values are empirical, and it may be that further work is needed to do better.

The chrominance values were calculated for each pixel in the region and were used to construct a color histogram with 16 bins. An intensity histogram was also created, having only 4 bins; both were used in searches.

5. Regions

Based on prior experimentation by ourselves and others (E.G. [9]) it is generally appreciated that statistical measures based on entire images are often less successful in characterizing the image in a search or matching context than the using same measurements based on subdivisions of the same image. When using regions, one of the five defined measurements is made on each defined region and collected into a larger set of features. We identified five distinct ways to break up an image into regions: *overall*, *rectangular*, *angular*, *circular*, and *hybrid*, and tried each of them, comparing the results. The shape of these regions is illustrated in Figure 1.

Overall

This is the *null* or *trivial* region, the entire image considered as a single region. This corresponds with the usual global techniques for image analysis and recognition. This is used, for example, in [13].

Rectangular

This is a first step towards regionalization of an image, and is simple to implement because an image is rectangular, and so are the regions. For the experiments described here, the image is broken into five vertical and horizontal parts. Features are then extracted from each of the 25 regions in the grid.

For example, if the image is 250 x 500, then each region is 50x100 pixels. there is no overlap between the regions, although there certainly could be. *Advantages*: fast, easy to implement. *Disadvantages*: not designed with rotation in mind.

Angular

Angular regions are wedge-shaped sections of the image radiating from the geometric center of the image (not the centroid. which is pixel based). We used an angular differential of 45 degrees, creating eight angular regions for our experiments. *Advantages*: all regions have some center and some outlying pixels; nice rotational properties. *Disadvantages*: more difficult (expensive) to implement than *rectangular* regions. Some angles are more accurately sampled than others.

Circular

Circular regions consist of concentric circles or rings beginning at the geometric center of the image, as nearly as possible. We used five rings for our experiments. The radius of the last ring is equal to the maximum of the largest row and column index. *Advantages*: rotationally invariant; using more rings is not really more costly. It is easy to weight the rings so that the image center is more important. *Disadvantages*: expensive to implement, pixels near the boundary are arbitrarily assigned to regions.

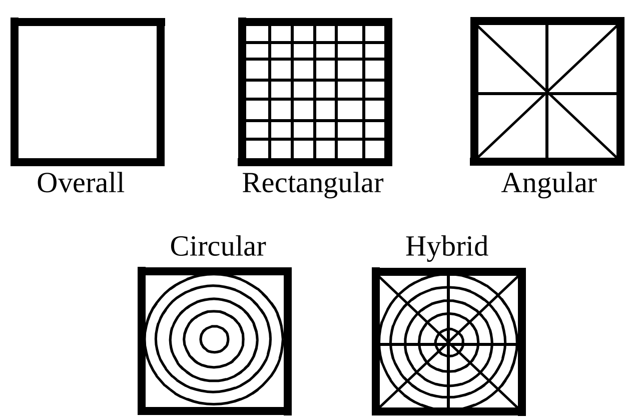
Hybrid

Hybrid regions represent a logical combination of angular and circular regions, as defined above. This is really just a two dimensional polar coordinate system. Both the concentric rings seen in circular regions and the radial segments of the angular regions are superimposed. In the experiments defined here, there were 8 angular regions and 4 circular ones, for a total of 32 regions.

6. Experiments

Our experimental database contains 782 images in 8 classes. Seven of the classes had 100 images, while the

Figure 1: The five types of sub-image used in the similarity search process.



final one had only 82. The *accuracy* A for each class is calculated as $A = 100\frac{c}{qn}$, where ***c*** is the number of correct (in-class) retrievals and ***n*** is the number of images in that class [4]. The symbol ***q*** represents the number of results that were returned. We are using a search engine evaluation scheme, which requires some explanation.

When a search engine is given a query, the resulting responses are ranked according to relevance, and are returned and displayed in that order. The number of responses on the page is ***q***, frequently 10 or so. In the results presented here ***q***=30.

There are many ways of reporting success in this kind of enterprise. We are suggesting that success is *the percentage of relevant responses on the first page of a typical query.* This is certainly a measure of success that would be quickly understood by anyone who uses web search engines frequently. Out of the ten responses reported on the first page of a response to a query, how many of them are really a match? When asked this question of text based queries, the average person would accept 3 successes, which they think of as typical; this is based on a recent casual poll of students and University staff.

Given this measure of success, the overall system was initially tested on the 782 images at our disposal. Eight tables, one for each class, are needed to convey all of the information resulting from the trials. Each image is queried against the database, and a table of success percentage with similarity algorithms occupying columns and region drawing methods as rows. Four of the tables are shown collected as Table 1, and it is plain to see that there is a significant variability in success among the classes tested. It is also plain that the method and region scheme that works best for one class does not necessarily work the best for some other.

What is wanted is a scheme that works best for all classes. Figure 2 shows representative samples of each class.

7. ALGORITHM FUSION

The experiments involved searching for each image amongst all of the other images, using every combination of similarity measure and region splitting method described above. This means that a similarity value has been calculated for all image pairs in the data collection. These can be sorted into a ranked list for each

Table 1: Region/Feature Accuracies for some classes

	Grey Sigma	Edge Direction	Edge Density	Boolean edge density	Hue Histogram	Intensity Histogram
			Accuracy for class 'beach' (%)			
Overall	12.8	18.3	9.6	11.6	23.7	18.5
Rectangular	13.2	20.7	9.6	10.2	22.1	12.7
Angular	12.7	15.1	9.6	9.6	23.7	14.9
Circular	20.0	12.5	13.1	15.2	23.4	20.1
Hybrid	22.1	21.4	5.8	14.3	25.5	12.4
			Accuracy for class 'horses' (%)			
Overall	20.3	22.3	13.7	12.9	79.3	34.0
Rectangular	47.7	11.1	44.8	38.7	90.8	49.5
Angular	23.9	16.9	25.6	24.7	85.9	43.7
Circular	41.0	20.3	26.6	25.3	85.9	38.6
Hybrid	42.2	13.3	29.0	37.8	86.8	45.8
			Accuracy for class 'dinosaur' (%)			
Overall	11.2	48.0	25.1	38.6	24.1	97.0
Rectangular	41.7	49.4	54.3	69.5	51.2	100
Angular	16.6	45.0	34.3	48.0	26.4	99.6
Circular	70.9	42.7	74.3	82.1	29.2	99.1
Hybrid	53.9	57.2	0.0	22.7	57.9	98.8
			Accuracy for class 'flower' (%)			
Overall	11.2	14.8	23.7	17.5	38.4	49.7
Rectangular	43.8	28.9	56.5	47.7	40.9	59.8

algorithm, in which the first image is the best (highest similarity value, most likely match). We can use these ranked lists as a means to vote for the best match. The method we have used in the past to do this is called a *Borda count* [2,7].

The problem encountered when attempting to merge ranked responses is as follows: given M rankings, each having N choices, which choice has the largest degree of support? For example, consider the following 3 classifier/ 4 class problem [11]:

C1: a b c d C2: c a b d C3: b d c a

This case has no majority winner; a, b and c each get one first place vote. The *Borda count* is an ancient scheme for resolving this kind of situation, in which each alternative is given a number of points depending on where in the ranking it has been placed. A selection is given no points for placing last, one point for placing next to last, and so on, up to **R**-1 points for placing first. In other words, the number of points (the weight) given to a selection is the number of classes below it in the ranking.

Other voting methods were attempted, such as the simple majority vote, weighted Borda count, and so on, but the simple Borda count appeared to provide the most robust solution. The overall results, using this algorithm combination methods, were as follows:

beach	28.78%	flower	81.67%
horses	86.37%	architecture	40.43%
dinosaur	98.97%	bus	58.78%
elephant	39.30%	mountain	26.90%
	Overall 56.95%		

This means that, in a web search having ten results per page, the first page would have 5-6 correct (directly relevant) matches to the query, on the average. This is better than our informal poll suggests is acceptable, and better than the same poll is being achieved now on text-based queries.

8. Comparison Against the Literature

We performed a formal comparison of our multiple algorithm method for image similarity search against two published methods: that of Rao[9] and that of Tico[13]. Both methods were implemented by us using the original papers as the correct description of the method; this means that there is a chance that the program we used to generate the results is somewhat different from the one used by the original authors.

The results were computed in the same was as for the previous experiment, and are tabulated in Table 2. In all cases but one our composite system gives better success rates than any of the others.

9. An Interesting Last-Minute Experiment

For the past years our research group has been working on specified techniques for censoring pornographic web sites so that public schools can use the web as a resource more comfortably. We have used many methods combined with voting schemes, but are especially interested in methods that look at the images and can classify them as *innocent* or *rejected (*I.E. nudity*)* based on content. After our experiments with the composite image search engine were success, we quickly put together a small set of data to see whether the engine could distinguish between these two classes. The innocent images were gather specifically for this test, and consisted of partly dressed and lightly dressed people, some posed and some in natural situations. The rejected images contained nudity and poses unsuitable for younger viewers.

The data set is too small to draw many conclusions at this time, but the results are promising. Within an hour we had five false rejections and no false acceptances - given ninety images in each class, this amounted to about 97% success. After some time adjusting parameters and carefully selecting a threshold, this was improved to a single false rejection, or over 99% success. This is almost

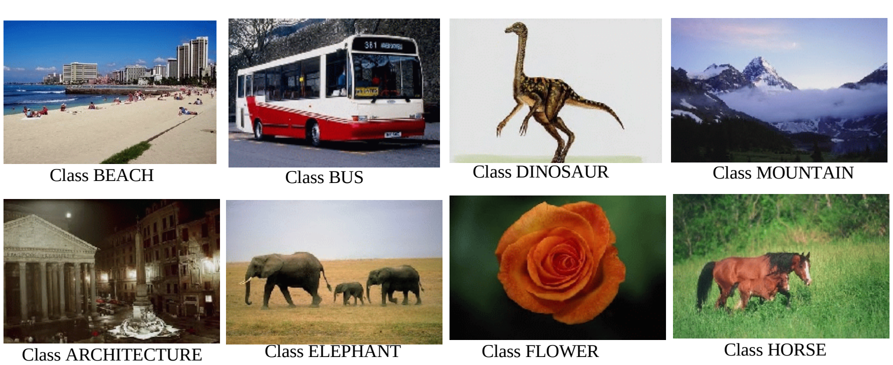


Figure 2: Examples of the image classes used in the similarity search experiments. There are 8 classes, each with 100 instances except the mountain class, which had only 82.

Table 2: Final results - Comparison Between Methods

Image Class	Correct retrievals (Rao[9]) %	Correct Retrievals (Tico[13]) %	Correct Retrievals (This work) %
Beach	27.7	25.6	28.8
Horses	89.0	68.3	86.4
Dinosaur	42.0	72.6	99.0
Elephant	20.0	24.7	39.3
Flower	46.4	51.3	81.7
Architecture	27.0	24.2	30.4
Bus	36.0	33.7	58.8
Mountain	26.0	19.9	26.9
TOTAL	39.5	40.4	57.0

certainly high, but when we add text features and other image censoring operations we have already developed, and given the ease with which the success rate was achieved, we expect a censor with well over 99% success this year, partly based on this work

10. Conclusions

We have described a small collection of fast methods for determining similarity between images, and have evaluated them on a set of 782 images. These algorithms were then combined, using a rank-based voting scheme, to produce a multiple algorithm system that gives overall results that are significantly better than two of the methods found in the literature.

The prototype system is completely functional, and we hope to have it installed in a publicly assessable system within the next few months. A more complete collection of the results can be found by contacting the authors.

11. Ackowledgements

We thank the Natural Sciences and Engineering Research Council of Canada for their financial support.

12. References

1. A. del Bimbo, *Visual Information Retrieval*, Morgan Kaufmann, San Francisco, 1999.
2. Jean-Charles de Borda, *Memoire sur les Elections au Scrutin*, **Histoire de l'Academie Royale des Sciences**, Paris, 1781.
3. J. Hafner, et. al., *Efficient Color Histogram Indexing for Quadratic Form Distance Functions*, **IEEE Transactions on Pattern Analysis and Machine Intelligence**, 19(7), Pp. 729-736, 1995.
4. H. Muller, S. Marchand-Mailler, and T. Pun, *The Truth About Corel - Evaluation in Image Retrieval*, **Proc. CIVR 2002**, Pp. 38-49, Springer-Verlag, 2002.
5. W. Niblack, R. Barber, W. Equitz, M.Flickner, E. Glassman, D. Petkovic, P. Yanker, *The QBIC Project: Querying Images by Content Using Color, Texture, and Shape*, **SPIE 1908, Storage and Retrieval for Image and Video Databases**, Feb. 1993
6. J. Parker, *Algorithms for Image Processing and Computer Vision*, John Wiley & Sons Ltd., New York, 1996.
7. J.R. Parker, *Voting Methods for Multiple Autonomous Agents*, **Proc. ANZIIS'96**, Perth, Australia, 1996.
8. Parker, J.R., *Histogram Methods For Scientific Curve Classification*, **SPIE Vision Geometry VI**, San Diego, July 28-29, 1997.
9. A. Rao, R. K. Srihari, and Z. Zhang, *Spatial color histograms for content-based image retrieval*, **Proceedings of IEEE International Conference on Tools with Artificial Intelligence**, pages 183--186, November 1999.
10. T. Seidl and .H. Kriegel, *Adaptable Similarity Search In Large Image Databases*, in **Content Based Image and Video Retrieval**, R. Veltkamp, H. Burkhardt and H-P. Kriegel eds., Kluwer, Publishers, 2001.
11. P.D. Straffin, Jr., *Topics in the Theory of Voting*, Birkhauser, Boston, 1980.
12. M. Swain and D. Ballard, *Color Indexing*, **International Journal of Computer Vision**, Vol. 7, Pp. 11-32, 1991.
13. M. Tico, T. Haverinen, and P. Kuosmanen. *A method of color histogram creation for image retrieval*, In **Proceedings of NORSIG**, 2000.

Rotation and Scale Invariant Color Image Retrieval Using Fuzzy Clustering

Shan Li, M.C. Lee
Computer Science & Engineering Department, Chinese University of Hong Kong, Hong Kong
Email: sli@cse.cuhk.edu.hk, mclee@cse.cuhk.edu.hk

Abstract

Color is one of the most important image features used in content-based image retrieval (CBIR). This paper proposes a robust and effective color feature known as ring-based fuzzy histogram (RFH), which reduces noise sensitivity in conventional color histograms (CCH) by using two stages of fuzzy clustering. By partitioning images into rings, RFH can capture an image's spatial information, and is robust to rotation and scale variations. This paper also presents details about the extraction of RFH. The fuzzy histogram extracted from the whole image is used to filter out highly dissimilar images during the image retrieval process, leading to a reduced query response time. Experiments results show that the proposed feature is, in general, compact; robust to changes due to images rotation or image scaling; and robust to illumination changes and quantization noise. It also outperforms CCH and other recently published proposal, such as fuzzy color histogram [6].

1. Introduction

Color is an important image feature for the task of content-based image retrieval. In the early 1990's, Swain and Ballard [14] were among the first to propose the use of conventional color histogram (*CCH*) in image retrieval. It maps the colors in an image into a discrete color space containing n colors. Color histograms are easy to compute, relatively robust to image distortions, (such as those due to rotation). However, there are also problems with the color histograms: First, the number of color bins needed in *CCH* is large and thus a comparatively large space is needed to store each image. Second, intersection-based similarity measurement leads to the sensitiveness to quantization errors and illumination changes (e.g., two colors will fall into different histogram bins even when they are very similar). Third, CCH doesn't include important details (such as spatial information) of images [6].

Some clustering methods had been employed to reduce the dimensionality of the color histogram. Examples here include hierarchical clustering, color naming system, supervised clustering, and uniform quantization.

To improve the robustness of the color histogram, [14] proposed a similarity measurement based on histogram intersection with the aim of eliminating the influence of background color in the match results. Although the method is robust to object occlusion and image resolution, it is still largely sensitive to changes in illumination. [7] assumed a linear spectral reflectance model to derive a set of illumination-independent moment invariants of color distributions. Other methods extended color indexing by using the distribution of color ratios. Invariance to illumination conditions is achieved on the assumption of locally constant illumination [4][10]. Though effective, their computation is time-consuming.

Trying to address both of two problems mentioned above, the method of Fuzzy Color Histogram (*FCH)* is proposed by applying fuzziness to color histograms [6]. Each pixel's color is associated to all the histogram bins through the function of fuzzy membership. The experiments showed that requiring less storage space, *FCH* is less sensitive to noise interference, such as illumination changes and quantization errors than *CCH*. However, this approach *FCH* simply ignored the quantization error introduced in the process of uniform quantization, which impaired its robustness to noise interference. Besides, it doesn't incorporate spatial information in the image feature, which leads to ambiguities when comparing similarities by histograms.

Common methods incorporate spatial information by dividing images to different blocks [2][5][13]. Yet such methods are not robust to image rotations. Other methods try to integrate spatial information without partitioning the image [8] [11] [12]. These methods typically have some special restrictions (eg. the number of representative colors [8], or the number of "sufficiently present" colors [12], should not be large). They are also usually time-consuming to implement. To our knowledge, there is still no commonly accepted method to incorporate spatial information with color features, while at the same time maintaining rotation and scale invariance.

In this paper, we propose a new image feature, called ring-based fuzzy histogram (*RFH*). Images are partitioned into concentric rings with varying radial distances first, in order to integrate spatial information while holding the property of rotation invariance. Two steps of fuzziness, fuzzy smoothening and fuzzy quantization, are introduced to *CCH*, to make the histogram more robust to noise interference. Experiments show that *RFH* generally outperforms *FCH* and *CCH* in the robustness to noise interference and in the subject discriminability.

This paper is organized as follows. In the next section, we present the details on the two steps of fuzziness in problem of color quantization. Section 3 describes the ring-partitioning method applied on the image, as well as the extraction process of *RFH*. Experimental results are provided in Section 4. Section 5 concludes the paper.

2. Fuzzy color quantization

To reduce the dimensionality of the resulting color histogram features, an appropriate method should be employed to quantize the color values. In this section, we describe a two-step approach to introducing fuzziness in problem of color quantization. First, the image is taken through an initial process of fuzzy color smoothening. Based on this, we perform fuzzy quantization using a perceptually-uniform color space, the $CIE(L^*, u^*, v^*)$ color space.

2.1. Fuzzy histogram smoothening

For color image in *RGB* space, there are usually 256 levels of colors in each channel. This number is not necessary for most color image recognition applications. Color levels are reduced from 256 levels/channel to 16 levels/channel through method of uniform quantization in our work. However, it could introduce quantization error during the process of quantization.

In order to lessen the quantization error, we first introduce fuzziness in our framework by the use of a fuzzy smoothening function on 4096(16*16*16)-bin RGB color histogram. For a given color level, we associate with it a fuzzy membership function *f*. Thus, given the *i*-th color level p_i, the membership function $f_i(j)$ defines the degree to which the *j*-th color level is similar to p_j, where the values in $f_i(j)$ are normalized to be in the range [0 1]. Expectedly, the degree of similarity should be inversely proportional to the inter-color distance between p_i and p_j. The natural choice for this function is the Gaussian smoothening function:

$$f_i(j) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{d(i,j)}{2\sigma^2}\right) \tag{1}$$

where, σ^2 is the variance, and $d(i,j)$ is a distance measure.

Typically, the distance measure is the *L2*-norm: $d(i,j) = \|p_i - p_j\|$. Given the membership function, the conventional color histogram *H* can now be smoothened as follows:

$$H'(i) = \sum_{j=1}^{n} H(j) f_i(j) \tag{2}$$

This is essentially a convolution operation. Thus, for image *M*, we can represent the fuzzy smoothened histogram as $H'(M) = H * f_i$, where * denotes the convolution operator.

The resulting histogram after the fuzzy smoothening has been termed "fuzzy paradigm-based histograms" in [15]. This approach has also been used by some authors for the purpose of reducing quantization noise [9]. We note that the above smoothening can be performed either on the *RGB* or on the CIE(L^*, u^*, v^*) space.

2.2. Fuzzy color clustering

To further improve efficiency, the histogram with 4096 bins can still be reduced with less bins. However, simply using fewer bins could lead to a significant loss in the color information. Therefore an appropriate quantization technique is required to reduce this potential loss, while still providing a more compact representation. We note two important observations in choosing a clustering method for the purpose of image retrieval.

First, a fine clustering method works better in a perceptually uniform color space, (such as the CIE (L^*, u^*, v^*) space used in the work); Second, color bins in *RGB* color histogram have a proportional spacing between them. And the proportional spacing between bins is lost after the nonlinear transformation from the *RGB* space to the CIE(L^*, u^*, v^*)space. Thus, uniform quantization (i.e., uniformly dividing the data equally to different clusters) is not appropriate in clustering colors in the CIE (L^*, u^*, v^*) space.

Fuzzy clustering method was shown to be effective for non-regularly distributed data points [1]. It was also shown to improve the performance of color image retrieval [3][6]. Therefore, we use Fuzzy C-Means (*FCM*) clustering method to cluster the color bins in the CIE (L^*, u^*, v^*) space. The *FCM* algorithm attempts to partition a finite collection of elements $X = \{x_1, x_2, ..., x_n\}$ into a finite collection of *c* fuzzy clusters with respect to some given criterion. A detailed description and implementation of the *FCM* algorithm can be found in [1].

2.3. Color quantization using fuzzy clustering

The procedure of color histogram quantization is described as follows:

1) Perform uniform quantization to the *RGB* color histogram, which reduces the color levels from 256 bins/channel to 16 bins/channel. We call this step *coarse quantization*.
2) Combine 3-D histogram to 1-D histogram and compute the index number of each bin in the 1-D histogram.
3) Using Gaussian smoothening function to smoothen the 4096-bin color histogram.
4) Convert the 4096 colors from *RGB* space to the CIE (L*,u*,v*) coordinates.
5) Cluster the 4096 color bins to *c* color clusters/bins by fuzzy quantization, using the *FCM* algorithm, where *c* is the required number of color clusters. We call this step *fine quantization*.

The 3D to 1D mapping in step 2 is performed using a simple relation:

$$f(r,g,b) = b \times 16^2 + g \times 16 + r \,, \tag{3}$$

where r, g, b are the respective color values in the 3D RGB space,. $f(r,g,b)$ is the index number of the bin. 16 levels in each of the 3 channels lead to 4096 different levels at the 1D representation

The choice of the parameter *c*, the needed number of clusters will typically be application dependent. In this work, we performed a performance comparison using different values of *c*, (*c*=32, 64, 128). Results show that retrieval performance is only marginally improved beyond *c* = 64, (see section on results).

In step 5, a mapping table between the 4096 CIE(*L*,u*,v**) values and the *c* color clusters is generated. It serves as a mapping between the 4096 *RGB* values and the *c* fuzzy color clusters. The mapping table is computed offline only once, and is represented as a partition matrix *U*:

$$U = \{u_{ij}, i = 1,2,...,c, j = 1,2,...4096\}\,, \tag{4}$$

where u_{ij} represents the degree of membership (or association) of the *j*-th color value (from 4096 possible RGB values) with the *i*-th color cluster (out of c clusters). Thus, each row of *U* represents a fuzzy membership function.

A quantized histogram (so called *fuzzy histogram*) could be computed directly from the conventional histogram, using simple matrix multiplication:

$$\underset{1\times c}{FH} = \underset{1\times 4096}{H} \times \underset{4096\times c}{U^T} \tag{5}$$

where, *FH* is an *c*-bin *fuzzy histogram*, *H* is vector representing 4096-bin normalized *RGB* color histogram Since normalized RGB color histogram *H* is scale invariant, the *fuzzy histogram FH* is also scale invariant.

3. Ring partitioning and ring-based fuzzy histogram

As mentioned in Section 2, a number of approaches attempted to integrate spatial information with color histograms by dividing images into sub-blocks and comparing the features extracted from the sub-blocks. Yet the features in these methods are not rotation invariant. In this section, we present a new partition method. The method is easy to implement. More importantly, the image features extracted from the resulting partitions possess the important property of rotation invariance.

3.1. Ring partitioning method

Image rotating about uncertain point by angle θ_0 can be considered as rotating about the center of the image frame by angle θ_0 and then shifting by an amount that depends on the position of the uncertain point. For simplicity, we only consider rotating about the centroid. Translation is a concern which is beyond the scope of the present work.

A number of rings are used to partition the image to sub-regions. The rings are concentric, centered at the center of the image frame. The areas between neighboring rings are the sub-regions based on which the image features will be extracted. In order to more easily partition the image circular wise, polar transformation is applied to each image. With polar transform, rotation in the Cartesian space can be performed as planar shifts in the polar coordinates.

3.2. Polar images

A polar image can be constructed as follows: for a given $L_1 \times L_2$ image $f(x,y)$, scan the image in two dimensions. First, scan *M* times along the angular direction, evenly distributed from 0° to 360°. Second, scan *M* times along the radial direction, evenly distributed from 0 to $N/2$, where $N = \max(L_1, L_2)$ (see Fig. 1).

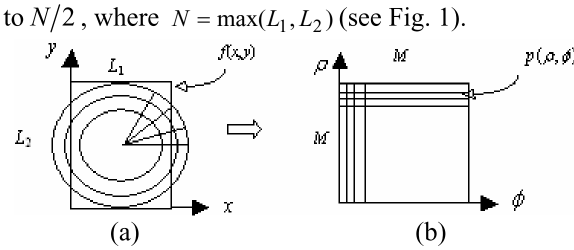


Figure 1. **The polar transformation: (a)** $L_1 \times L_2$ **image** $f(x,y)$ **in Cartesian coordinates; (b)equivalent** $M \times M$ **polar image** $p(\rho,\phi)$ **in polar coordinates.**

For the example of $f(x,y)$ in (a), we have $L_1 < L_2 \Rightarrow N = L_2$. For the regions of rings that fall out of the square area of $f(x,y)$ in (a), theirs color values are set to be 0 in the corresponding positions in the polar image, $p(\rho,\phi)$.

Formally, $p(\rho,\phi)$ can be computed as:

$$p(\rho,\phi) = f(\lfloor N/2 + \rho\cos(2\pi\phi/M)\rfloor, \lfloor N/2 - \rho\sin(2\pi\phi/M)\rfloor) \quad (6)$$

where $\lfloor x \rfloor$ represents the smallest integer that is not greater than x. Note that rotation in Cartesian space corresponds to a simple cyclic column shift in the polar space.

3.3. Ring-based fuzzy histogram

We partition the polar image to several sub-regions (or layers), where each region/layer is composed of the same number of consecutive rows in the polar image. Rotation and scale invariance can be achieved by computing fuzzy histograms from each sub-region in the polar image. We call these histograms *Rring-based Fuzzy Histograms* (*RFH*). Essentially, *RFH* is a set of fuzzy histograms defined as follows:

$$RFH = \{FH_0, FH_1, FH_2, ..., FH_m\}, \quad (7)$$

where, FH_0 is the c-bin *fuzzy histogram* of the whole polar image $p(\rho,\phi)$. $FH_i(i = 1,2,..,m)$ is the c-bin *fuzzy histogram* of the i-th sub-region in the polar image, m is the total number of sub-regions.

There are two kinds of *RFH* features: the global features FH_0, and the regional features $FH_i(i = 1,2,..,m)$.

1) Global features: FH_0 is the first component of *RFH*. This is a global feature that describes the color statistics from the entire image. For a given query image q, and a database image d, we define below a normalized distance $D(q,d)$ between them based on FH_0:

$$D(q,d) = \frac{\sum_{j=1}^{c} \left| FH_0^q(j) - FH_0^d(j) \right|}{1 + \sum_{j=1}^{c} [FH_0^q(j) + FH_0^d(j)]} \quad (8)$$

where j is the bin position in the c-bin *fuzzy histogram*.

We use the global feature FH_0 to filter out images that differ significantly from the query – in terms of their general color statistics. Since spatial information about the image is maintained in $FH_i(i = 1,2,..,m)$, in a sense, using FH_0 in this way can be viewed as using the image's general color information as a pre-filter before using detailed spatial information for further processing. This proved to be time saving during retrieval, and also improves the retrieval accuracy.

2) Region features: After pre-filtering the database images, $FH_i(i = 1,2,..,m)$ are then used to further prune out dissimilar images. To decide the value of m, there are several factors taken into consideration. First, with a bigger value of m, color image is partitioned by more rings, thus more spatial information is kept in FH_i. This is helpful to filter out those images that are visually dissimilar, but have similar global color statistics. Second, if the value of m is too big, each area of the ring-region gets small. Thus a small translation of the color image will have a significant influence on the contents of those ring-regions. As a result, the *RFH* from these areas will be less robust to small image translations. Third, FH_i with a bigger m surely takes more storage space and more computing time. Consider the 3 factors, an optimal number of rings needs to balance the positive and negative effects listed ahead. The value of m will be decided in the experiments.

The overall normalized distance between the query image I_q and the database image I_d is as follows:

$$D(I_q, I_d) = \sum_{i=1}^{m} w_i \frac{\sum_{j=1}^{c} \left| FH_i^q(j) - FH_i^d(j) \right|}{1 + \sum_{j=1}^{c} [FH_i^q(j) + FH_i^d(j)]} \quad (9)$$

where $\sum_{i=1}^{m} w_i = 1$, and w_i is the weighting factor for the i-th sub-region of the polar image. The choice of w_i will depend on the importance associated with each sub-region. In our experiments, we assume that each sub-region provides roughly equal contribution to the similarity measure, and hence use $w_i = 1/m$.

4. Experiments

4.1. Construction of test databases

Three test image databases were constructed to evaluate the performance of the proposed *RFH* on three grounds: general discrimination ability, rotation invariance, and robustness to illumination changes in the retrieval of similar images. The databases are called *general database, rotation database* and *illumination database* respectively. With each of the three databases, all the 1,500 images with different subjects are used as queries. Similar images are retrieved by comparing the query with the database images. Retrieval results from all queries in one database contribute to the average performance of all the queries in this database.

4.2. Measurement of retrieval performance

We use a precision-recall (*P-R*) graph to evaluate the retrieval performance of different image descriptors in rotation and scale test databases. Precision *P*, and recall *R* are defined respectively as follows:

$$P = \frac{|C_q|}{|B_q|}\ ;\ R = \frac{C_q}{D_q} \qquad (10)$$

For a given query, we can set the recall to different values, and then calculate the corresponding precision values. From all query results, we plot their average precision value for each recall in the *P-R* graph.

4.3. Results

1) General Database: To check the effect of the number of rings on the performance of *RFH*, we carried out a good number of experiments on the general database, using different number of rings to partition the color images. Fig.2 shows the overall *P-R* performance comparison.

The optimal number of rings will typically be application dependent. From Fig. 2, *RFH* with 4 rings produced the best overall *P-R* performance. Thus, for the remaining experiments, we fixed the number of ring partitions to 4. Due to the fact that similar images in the database are often in different size, the ability to retrieve these images demonstrates that *RFH* is robust to scale changes of images (i.e. scale invariant).

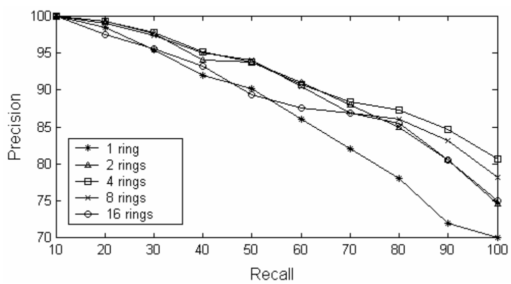


Figure 2. **Performance of *RFH* using different number of rings to partition the images.**

From Fig. 2, *RFH* has the poorest *P-R* performance when using just one ring. Since this corresponds to using only the global feature (i.e. no regional information is used), it illustrates the importance of spatial information in color-based image retrieval.

To test the subjective discriminability of *RFH*, *FCH* and *CCH*, we perform a series of experiments on general database using the three features. For each pre-set recall value, average precision value of different feature is listed in Table 1.

Table 1. ***P-R* performance of *RFH*, *FCH* and *CCH*. For *RFH* and *FCH*, the number of bins refers to the stage of fine-grained (fuzzy) quantization (i.e. the number of fuzzy clusters *c*).**

Recall (%)	10	20	30	40	50	60	70	80	90	100
RFH (32 bins)	100	98.5	96.7	94.0	91.3	90.5	88.5	85.9	83.4	78.5
RFH (64 bins)	100	99.3	97.8	95.1	93.8	92.7	88.4	87.3	85.6	81.6
RFH (128bins)	100	99.5	98.2	97.4	94.5	92.3	90.9	87.7	86.5	82.3
FCH (32 bins)	100	97.4	96.7	93.7	92.5	90.9	87.0	84.5	83.7	77.4
FCH (64 bins)	100	98.2	97.9	95.0	92.6	90.0	87.5	86.9	84.0	79.3
FCH (128 bins)	100	99.4	97.8	94.4	91.5	90.7	87.7	87.1	84.2	80.3
CCH (32bins)	100	97.7	92.1	89.0	86.4	80	77.5	75.5	70.4	65.5
CCH (64 bins,)	100	98.2	93.4	90.3	86.6	84.5	83.8	76.5	73.4	70.9
CCH (128bins)	100	98.9	94.5	91.7	88.6	87.9	85.1	82.1	79.5	77.4

Several conclusions can be drawn from the table above. First, under the same number of bins, *RFH* achieves better *P-R* performance than *CCH* and *FCH*. This implies that by incorporating spatial information, *RFH* is more accurate at retrieving similar images. Second, performance curves for *RFH* and *FCH* are closer than the results with *CCH*. This implies that, *RFH* and *FCH* are less sensitive to changes in the number of bins (i.e. clusters) used. As amount of quantization error is mostly dependent on the number of bins used, the implication is that *RFH* and *FCH* features are more robust to image noise and quantization errors than *CCH*. Third, for *RFH*, retrieval performance with 64 bins is close to the performance with 128 bins. For the rest of this section, we report results for 64 bins.

2) Rotation Database: We also compare the performance of *RFH* with histogram features that are based on block partitioning. To avoid the influence of factors other than rotation, for the block-based color features, we also extract fuzzy histograms from the partitioned blocks. Besides, we compare the performance of *RFH* with that of *FCH* on rotation test database.

Fig.3 presents the overall precision-recall charts for the three features using the rotation databases.

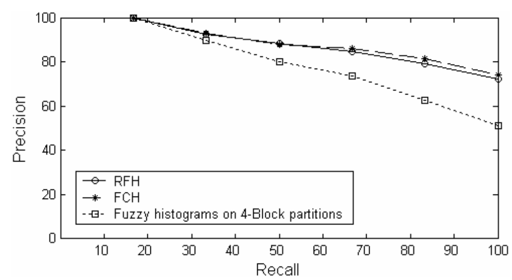
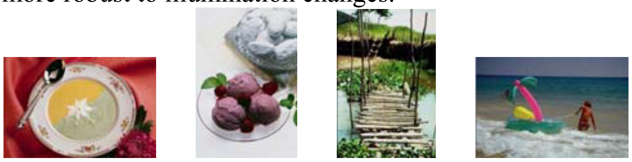


Figure 3. **Precision-recall plots using the three approaches to integrating spatial color features.**

When compared with the feature based on block partitioning, *RFH* with ring partitioning and *FCH* can produce higher precisions for many of the different recall values. The difference in performance increases with larger recall values. We can tell from above figure that *RFH* is robust to image rotations, and its robustness to image rotation is close to that of *FCH*.

3) Illumination Database: To compare the robustness of *RFH*, *CCH* and *FCH* to illumination variations, we performed tests on the illumination database by using the 3 features. For each feature, all database images are sorted based on their Euclidean distance to a query image. Fig.4 shows four randomly selected query images. Table 2, 3 and 4 list the ranks of the corresponding 10 images when using respective queries.

Comparing the retrieval results by using different features, similar images rank notably higher in Table 2 than in Table 3 and Table 4, which implies that *RFH* is more robust to illumination changes.



(a) (b) (c) (d)

Figure 4**. Query images from the illumination database**

Table 2. **Retrieval results by *RFH***

Original images	Ranks of the corresponding 10 images									
a	2	3	4	5	6	8	10	17	22	35
b	2	3	5	6	8	10	14	20	44	78
c	2	3	4	5	6	10	19	22	66	110
d	4	6	10	14	17	20	22	40	71	157

Table 3. **Retrieval results by *FCH***

Original images	Ranks of the corresponding 10 images									
a	2	3	4	6	7	10	14	17	20	60
b	2	4	5	6	11	14	17	28	66	121
c	2	3	4	5	6	9	17	23	73	127
d	4	7	13	14	19	26	32	41	96	178

Table 4. **Retrieval results by *CCH***

Original images	Ranks of the corresponding 10 images									
a	2	3	4	5	11	16	17	25	39	173
b	2	3	4	7	9	15	25	57	120	270
c	3	4	7	8	13	19	37	47	184	323
d	4	5	11	13	21	24	41	91	124	197

5. Conclusion

In this paper, we propose using the image feature known as *RFH*. To improve robustness, fuzziness is introduced in the histogram generation process in two steps. First, fuzzy smoothening is applied to the image histogram after coarse quantization, reducing the quantization errors introduced during the process of coarse quantization. At a later stage of fine quantization, *FCM* algorithm is used to cluster the histogram bins (4096 bins in the work) into a relatively low dimensional space. To integrate spatial information to the feature while holding the property of rotation invariance, ring-partitioning is applied to polar-form images and *RFH* is then constructed as a set of fuzzy histograms for the ring-partitions. To speed up the query process, the fuzzy histogram of the whole polar image is used as a global feature to filter out the highly dissimilar images, before performing the more time-consuming analysis of the region-based fuzzy histograms.

Experiment results show that, with introduction of two stages of fuzziness, *RFH* is more robust to noise and illumination changes than *FCH* and *CCH*; with spatial information integrated, *RFH* outperforms *FCH* and *CCH* in subject discriminability; with the ring-partitioning method, *RFH* is more robust to image rotations than traditional block-partitioning method.

Acknowledgement

The work reported in this article has been supported in part by the Hong Kong Research Grants Council under CUHK4377/02E.

6. References

[1] Bezdek J.C., Pattern Recognition with Fuzzy Objective Function Algorithms, New York: plenum,1981

[2] Dimai A., “Differences of global features for region indexing,” Swiss Federal Inst. Technol., Lausanne, Tech. Rep. 177, 1997

[3] Eschrich S., Jingwei Ke, Hall L.O. and Goldgof D.B., “Fast accurate fuzzy clustering through data reduction,” IEEE Transactions on Fuzzy Systems, vol. 11(2), pp.262-270, April 2003

[4] Funt B.V. and Finlayson G.D., “Color constant color indexing,” IEEE Transactions on Pattern Analysis and Machine Intelligence., vol.17(5), pp. 522-529, 1995

[5] Gevers T. and Smeulders A.W.M., “The PicToSeek WWW image search system,” Proceedings, IEEE International Conference on Multimedia Computing and Systems, vol. 1, 7-11 pp: 264 -269, June 1999

[6] Han J. and Kai-Kuang Ma, “Fuzzy Color Histogram and Its Use in Color Image Retrieval,” IEEE Transaction on Image Processing, vol. 11(8), pp. 944 -952, August, 2002.

[7] Healey G. and Slater D., “Global color constancy: recognition of objects by use of illumination-invariant properties of color distributions,” J. Opt. Soc. Amer. A, vol. 11, pp. 3003-3010, Nov. 1994.

[8] Hsu W., Chua T.S. and Pung H.K., “An integrated color–spatial approach to content-based image retrieval”, Proceedings, ACM Multimedia Conference, pp. 305-313, 1995.

[9] Kautsky J., Nichols N.K., and Jupp D.L.B., “Smoothed histogram modification for image processing”, Computer Vision, Graphics and Image Processing, 26, 271-291, 1984.

[10] ayar S.K. and Bolle R.M., “Reflectance Based Object Recognition,” Int’l J. Computer Vision, vol. 17(3), pp. 219-240, 1996.

[11] Pass G. and Zabih R., “Histogram refinement for content-based image retrieval,” in Proc. 3rd IEEE Workshop Applications Computer Vision, Dec. pp. 96-102, 1996.

[12] Smith J.R. and Chang S.F., “Tools and techniques for color image retrieval,” Proc. SPIE, vol. 2670, pp. 1630-1639, Feb. 1996.

[13] Stricker M. and Dimai A., “Spectral covariance and fuzzy regions for image indexing,” Mach. Vis. Applicat., vol. 10, pp. 66-73, 1997

[14] Swain M.J. and Ballard D.H., “Color indexing,” Int. J. Comput. Vis., vol. 7(1), pp.11-32, 1991

[15] Vertan C., and Boujemaa N., “Embedding fuzzy logic in content based image retrieval”, Proceedings, 19th International Meeting of North American Fuzzy Information Processing Society, NAFIPS 2000, Atlanta, 13-15 Jul, 2000.

An Image Watermarking Procedure Based on XML Documents

Franco Frattolillo, Salvatore D'Onofrio
Research Centre on Software Technology, Department of Engineering,
University of Sannio, Benevento, Italy

Abstract

This paper presents a watermarking procedure for JPEG images based on the use of protected XML documents. The procedure enables the copyright owner to insert a distinct watermark code identifying the buyer within the distributed images. Furthermore, to increase the security level of the procedure, the watermark is repeatedly embedded into an image in the DCT domain at different frequencies and by exploiting both block classification techniques and perceptual analysis. The embedded watermark is then extracted from an image according to the information contained in a protected XML document that is associated to the image.

1. Introduction and Motivations

Digital watermarking can be considered as one of the main technologies to implement the copyright protection of digital contents distributed on the Internet [1]. To this end, many watermarking procedures adopt "blind" insertion schemes and are based on fingerprinting techniques that enable the copyright owner to insert specific "anticollusion codes" able to identify the buyer within any copy of content that is distributed [12]. The main aim is to make it possible to establish if a user is illegally in possession of a content as well as who has initially bought and then illegally shared it via, for example, peer-to-peer network applications [1]. However, "nonblind" watermarking schemes are typically considered more robust then blind ones [8, 12]. Unfortunately, differently from blind ones, nonblind schemes need the original digital contents in order to be able to run the watermark extraction algorithms on the corresponding pirated copies. This is considered a drawback particularly for watermarking procedures that aim at being adopted in a web context, because such procedures force the distinct web entities involved in "identification and arbitration" protocols [8] to exchange unprotected, high dimension digital contents through the insecure communication channels characterizing the Internet [6].

This paper presents a web oriented watermarking procedure for JPEG images based on the use of protected XML documents [5]. The procedure enables the copyright owner to insert a distinct code identifying the buyer within each copy of the distributed images. Furthermore, to increase the procedure security and robustness level, the watermark is repeatedly embedded into an image in the DCT domain at different frequencies and by exploiting both block classification techniques and perceptual analysis. The embedded watermark is then extracted from an image according to the information contained in a protected XML document that is associated to the image. Thus, the usual security and robustness levels characterizing the nonblind watermarking schemes can be achieved without requiring unprotected, high dimension images to be exchanged in the Internet whenever the watermark extraction has to be performed. In fact, keeping XML documents protected or securely exchanging them in a web context results nowadays in being much easier than securely transferring high dimension images or having to manage their secure storaging at distinct web entities. Moreover, using the XML technology makes it also easier to automate the document access in a web context, since XML is a standard technology well supported by the Java world, and standard document parsers, such as SAX and DOM parsers, are freely available.

The paper is organized as the follows. Section 2 describes the proposed watermarking procedure. Section 3 shows how the information contained in the XML documents associated to the protected images can be exploited in the watermark extraction process described in Section 4. Section 5 reports on some experimental results. Section 6 reports conclusion remarks.

2. The Watermarking Procedure

The proposed procedure makes it possible to insert into a JPEG image a binary code represented by a sequence $\mu \in \{0,1\}$ and able to unambiguously identify a user. The sequence μ, whose length is denoted as n_μ, is repeatedly embedded into the image in the DCT domain at different frequencies, denoted as $\gamma_1, \gamma_2 \ldots \gamma_f$. In particular, since the coefficients in each 8×8 DCT block of an image have a frequency value associated with them, a γ value identifies an entry in such blocks, and so, it can range from 1 to $8^2 = 64$. Furthermore, to increase the security and robustness level of the procedure, the watermark insertion is assumed to be carried out at low, middle and high frequencies chosen on the basis of the image to watermark.

In principle, all the DCT coefficients at a given frequency could be modified by a value representing a watermark information. However, in the proposed procedure, the "perceptual capacity" of the coefficients belonging to the luminance DCT blocks is preliminarily estimated by exploiting both block classification techniques and perceptual analysis. In fact, the block classification techniques [3, 9] are applied to indicate the bests DCT coefficients that can be altered without reducing the visual quality. They classify each luminance DCT block with respect to its energy distribution by using four classification masks. The possible types of classified blocks are "flat", "diagonal edge", "horizontal edge", "vertical edge" and "textured block". The result of this procedure is a first selection of DCT coefficients whose modification has a minimal or no impact to the perceptual quality of the image.

The perceptual analysis is then applied to calculate the "just noticeable difference" (*jnd*) values for the DCT coefficients [7, 10, 11]. Such values are the thresholds beyond which any changes to the respective coefficient will most likely be visible. Therefore, let $X_{b_m}(\gamma)$ denote the DCT coefficient at the frequency γ in the block b_m, and let $JND_{b_m}(\gamma)$ denote the *jnd* value calculated for the $X_{b_m}(\gamma)$ coefficient. $JND_{b_m}(\gamma)$ can be calculated as:

$$JND_{b_m}(\gamma) \approx \max\left\{C_{b_m}(\gamma), |C_{b_m}(\gamma)|\, E_{b_m}(\gamma)^g\right\} \tag{1}$$

where $C_{b_m}(\gamma)$ represents the perceptual threshold of the contrast masking and is expressed as:

$$C_{b_m}(\gamma) = \max\left\{t_{b_m}(\gamma), |X_{b_m}(\gamma)|^h\, t_{b_m}(\gamma)^{1-h}\right\} \tag{2}$$

$E_{b_m}(\gamma)$ is the entropy value calculated over the eight neighbors of the $X_{b_m}(\gamma)$ coefficient [7, 10] and can be approximated by the following expression:

$$E_{b_m}(\gamma) \approx X_{b_m}(\gamma) - u_{b_m}(\gamma)q(\gamma) \tag{3}$$

In (1) g is assumed equal to 0.5, while in (2) h is assumed equal to 0.7 and $t_{b_m}(\gamma)$ is equal to $t(\gamma)(X_{b_m}(1)/X(1))$, where $X(1)$ is a DC coefficient corresponding to the mean luminance of the display, while $X_{b_m}(1)$ is the DC coefficient of the block b_m. In fact, $t(\gamma)$ can be approximated by the value $q(\gamma)/2$, where $q(\gamma)$ represents the coefficient of the quantization matrix corresponding to the frequency γ [10]. Finally, in (3) $u_{b_m}(\gamma)$ is equal to $round(X_{b_m}(\gamma)/q(\gamma))$.

The insertion procedure at a given frequency γ assumes that each bit of the user sequence μ is inserted into a given image by altering a pair of DCT coefficients associated to the frequency γ and chosen among the ones previously selected by applying the block classification techniques and perceptual analysis. In particular, the "choice rule" states that two DCT coefficients are allowed to belong to a same pair only if they have similar values. Consequently, if μ is n_μ bit long, the process that selects the DCT coefficients at the frequency γ has to choose at least $2 \cdot n_\mu$ coefficients. Moreover, if f insertion frequencies are chosen, the total number of DCT coefficients to select are $2 \cdot n_\mu \cdot f$.

To insert the bits of a user sequence μ into an image, the "encoding function" K has to be defined within the watermarking procedure. K defines an encoding rule by which the bits 0 and 1 are translated to the symbols belonging to the alphabet composed by $\{\nearrow, \searrow\}$, respectively called the *up* symbol and the *down* symbol. Thus, a user sequence $\mu \in \{0,1\}$ is translated to a corresponding sequence of symbols $\sigma \in \{\nearrow, \searrow\}$ depending on the function K. For example, the user sequence $\{01101\ldots\}$ is translated to the sequence of symbols $\{\nearrow\searrow\searrow\nearrow\searrow\ldots\}$, if the function K associates the *up* symbol to 0 and the *down* symbol to 1.

Let μ be a user sequence, and let σ be the corresponding sequence of symbols obtained by applying a K function. Let γ_1, γ_2 ... γ_f be the insertion frequencies. Let $W_{b_m}(\gamma_i)$ denote the watermarked DCT coefficient at the frequency γ_i in the block b_m. A symbol of σ is inserted into a pair of DCT coefficients belonging to the blocks b_m and b_n, at the frequency γ_i, by the following expressions:

$$\begin{cases} W_{b_m}(\gamma_i) = X_{b_m}(\gamma_i) - JND_{b_m}(\gamma_i) \\ W_{b_n}(\gamma_i) = X_{b_n}(\gamma_i) + JND_{b_n}(\gamma_i) \end{cases} \quad \text{to insert } \nearrow$$

$$\begin{cases} W_{b_m}(\gamma_i) = X_{b_m}(\gamma_i) + JND_{b_m}(\gamma_i) \\ W_{b_n}(\gamma_i) = X_{b_n}(\gamma_i) - JND_{b_n}(\gamma_i) \end{cases} \quad \text{to insert } \searrow$$

In fact, since the "choice rule" imposes that $X_{b_m}(\gamma_i) \approx X_{b_n}(\gamma_i)$ for each selected pair of DCT coefficients, the insertion process attempts to maximize the difference existing between the coefficients of the pair according to the direction specified by the insertion symbol and by an amount that should not compromise the final visual quality of an image. Therefore, the insertion process should be carried out according to the following rules:

1. The insertion frequencies should be evenly distributed among the low, middle and high frequencies, and should be chosen so that attacks characterized by a filtering behavior on an image would end up reducing its final visual quality. This can be achieved by selecting the frequencies characterized by high spectrum values, which, if filtered, can impair the image.

2. At each insertion frequency, the pairs of the selected DCT coefficients should belong to spatial regions that cannot be cropped without impairing the image.

Once the symbols of the sequence σ have been inserted into the image at the chosen frequencies, in order to increase the security and robustness level of the watermarking procedure against collusion and averaging attacks, it is necessary to hide the modifications made to the DCT coefficients of the image. In fact, let γ_1, $\gamma_2 \dots \gamma_f$ be the insertion frequencies chosen for the image, and let $\Sigma(\gamma_i)$ denote the sequences of the pairs of DCT entries $(b_m(\gamma_i), b_n(\gamma_i))$ that have been involved in the watermarking process for a given frequency γ_i, $\forall i = 1 \dots f$. It is worth noting that both the set of the frequencies γ_i and the sets $\Sigma(\gamma_i)$ are always the same for all the copies of a given image to protect. Consequently, the DCT coefficients modified at the different insertion frequencies remain the same for all the copies of the image. Therefore, in order to prevent malicious users from individuating the DCT coefficients modified by the insertion process, the *jnd* values modulated by a binary pseudo-noise sequence $\rho \in \{-1, 1\}$ have to be added to all the unmodified DCT coefficients of a watermarked image. This addition is carried out by the following expression:

$$X_{b_k}(\gamma_i) = X_{b_k}(\gamma_i) + \alpha_k \rho_k JND_{b_k}(\gamma_i),$$
$$(i \neq 1 \dots f) \text{ or } (i = 1 \dots f \text{ and } b_k \notin \Sigma(\gamma_i))$$

where $0 < \alpha_k < 0.5$ is a randomly varied amplitude factor.

3. The XML Documents

The capability of both repeatedly embedding a user code at different frequencies and hiding the watermarked DCT coefficients can make the proposed procedure almost secure against the most common filtering, corrupting, removal, averaging and collusion attacks. However, the characteristics of the insertion process could make the procedure vulnerable to geometric attacks. Therefore, to increase the robustness level of the procedure against such attacks, the attacked images should be geometrically re-synchronized before carrying out the watermark extraction.

To detect the most common geometric distortions applied to a watermarked image without having to use complex re-synchronization techniques, the proposed procedure makes use of some information about the image, which is assumed to be stored in a protected XML document associated to the image. This allows the information to be stored in both textual and quantitative form. The textual information can individuate and describe some evident and significant "feature points" and boundary segments of the image. The quantitative information can provide the original dimensions of the image, the coordinates of the feature points and selected boundaries, some Fourier descriptors and statistical moments of K-point digital boundaries, as well as the eigenvectors and eigenvalues of some well-defined regions of the image. Thus, inverse geometric transformations can be performed on the image in order to restore it before the watermark extraction [9].

Therefore, the XML document associated to each image to protect has to include: (1) the insertion frequencies γ_1, $\gamma_2 \dots \gamma_f$; (2) the sets $\Sigma(\gamma_i)$, $\forall i = 1 \dots f$; (3) the encoding function K; (4) some information about the original image, which can well and synthetically characterize the image and can be exploited to individuate possible geometric modifications performed on the image. Thus, when a pirated image is found in the market, a trusted third party (TTP) delegated to run the identification and arbitration protocols can retrieve the XML document associated to the image from the image's copyright owner in a protected and ciphered form. Then, the TTP can extract the code identifying the original buyer of the image from the pirated copy. This way, the image's copyright owner and the TTP are the sole web entities that are allowed to access the XML document associated to the image. This means that the security level achieved by the proposed procedure closely depends on the capabilities of both the copyright owners and TTPs of keeping the XML documents protected [4].

4. The Watermark Extraction

The first operation to perform before carrying out the watermark extraction from a given protected image is its geometric re-synchronization. Therefore, let Figures 1(a) and 1(b) be the watermarked and the attacked versions of the "Lena" image. To carry out the geometric re-synchronization, it is necessary to exploit the information stored in the XML document associated to Lena in order to build a reference picture (Figure 1(e)) whose dimensions coincide with the ones of the original image (Figure 1(a)). Then, the feature points connected by segments and specified by the XML document are to be reported on the picture (Figure 1(e)). These points have been originally determined on the watermarked Lena (Figure 1(c)) and are: the eyes of Lena, whose coordinates are (x_2, y_2) and (x_3, y_3); the tip of her hat, specified by (x_4, y_4); the right end of her mouth, specified by (x_1, y_1). The coordinates are referred to the X and Y axes, and the dimensions of the watermarked Lena are respectively d_x and d_y (Figures 1(c) and 1(e)).

The successive operation consists in reporting the feature points connected by segments on the attacked version of Lena (Figure 1(d)), which is a scaled and 45^o rotated version of Lena (Figure 1b). To this end, it is worth noting that the feature points can be reported on the image solely starting from the textual description provided by the XML document associated to the image (Figures 1(d) and 1(f)). This entails a natural approximation in individuating the feature points on the attacked image, which can determine errors in the geometric re-synchronization process involving Figures 1(e) and 1(f). However, the preliminary tests, conducted also on other images available on the web, have shown that the proposed procedure is robust with respect to such approximations. In fact, the procedure has been able to ensure a correct watermark extraction provided that

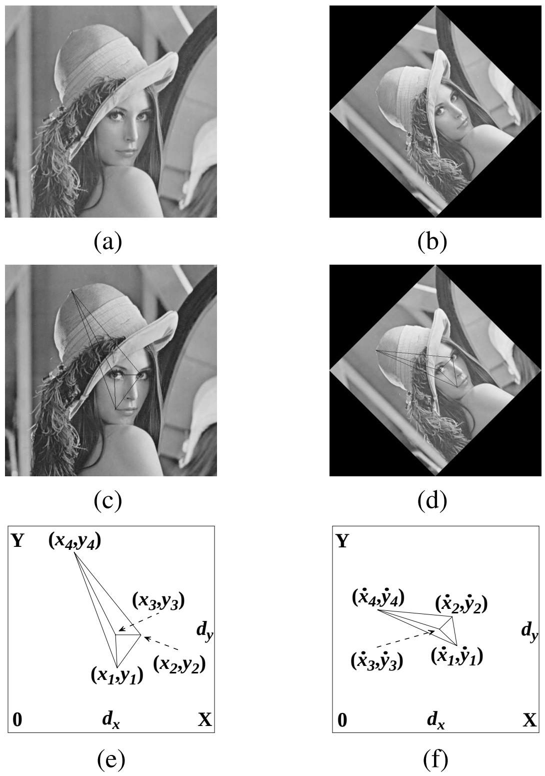


Figure 1. The geometric re-synchronization of "Lena".

the rotation degrees and scale factors are determined with approximations in the range of $\pm 6\%$. However, these limits have never been exceeded in the conducted, practical tests. To this end, Figure 2 shows the result of the worst re-synchronization performed on the scaled and 45^o rotated version of the watermarked Lena, which has not anyway prevented a correct watermark extraction, as reported in Section 5. In fact, the imperfect re-building of Lena essentially affects the outer regions of the image, i.e. the regions that do not, and should not, host watermark information.

After the geometric re-synchronization of the attacked image, the watermark extraction can be carried out. In particular, for each insertion frequency γ_i, the pairs of coefficients specified by the DCT entries $(b_m(\gamma_i),b_n(\gamma_i))$ belonging to the set $\Sigma(\gamma_i)$, with $i = 1 \dots f$, have to be examined. Therefore, let $\hat{W}_{b_m}(\gamma_i)$ and $\hat{W}_{b_n}(\gamma_i)$ be the two coefficients of a pair belonging to $\Sigma(\gamma_i)$. To extract the watermark symbol they host, the following expression has to be calculated:

$$\begin{cases} \hat{W}_{b_m}(\gamma_i) - \hat{W}_{b_n}(\gamma_i) > 0 \implies \searrow \text{is extracted} \\ \hat{W}_{b_m}(\gamma_i) - \hat{W}_{b_n}(\gamma_i) < 0 \implies \nearrow \text{is extracted} \end{cases} \quad (4)$$

Then, the extracted symbol is translated to a bit depending on the encoding function K.

After the watermark extraction, f user sequences $\hat{\mu}_i$ result in being re-built, one for each insertion frequency γ_i.

Figure 2. The watermarked "Lena" and its worst re-synchronized version.

Attack	f	$ber(\%)$	PSNR
JPEG 60	3	3.78	31.15
JPEG 60	6	3.61	29.96
JPEG 60	9	3.13	27.71
JPEG 40	3	5.77	26.51
JPEG 40	6	4.79	25.37
JPEG 40	9	4.53	24.92
Add Noise 5	3	1.95	25.79
Add Noise 5	6	1.67	23.44
Add Noise 5	9	1.2	22.13
Sharpening	3	1.86	33.33
Sharpening	6	1.47	31.02
Sharpening	9	1.35	30.92
Median 3	3	1.67	30.79
Median 3	6	1.34	30.12
Median 3	9	1.13	29.89
Median 9	3	1.99	24.33
Median 9	6	1.85	24.01
Median 9	9	1.63	23.88
Rotating 45°	3	3.85	23.45
Rotating 45°	6	3.12	22.91
Rotating 45°	9	2.61	22.7

Table 1. The results of some Stirmark attacks.

Therefore, let $\mu(j)$ denote the j-th bit of μ. $\mu(j)$ can be derived from the sequences $\hat{\mu}_i$ by the expression:

$$\mu(j) \equiv 1 \iff \frac{\sum_{i=1}^{f} \hat{\mu}_i(j)}{f} > 0.5, \quad \forall j = 1 \dots n_\mu$$

5. Experimental Results

To evaluate the security and robustness of the proposed watermarking procedure, some relevant attacks have been performed on many different images by using the widely accepted tool "Stirmark". However, for the sake of brevity, only the results obtained on Lena are reported. Furthermore, Stirmark does not perform "collusion attacks" [12]. So, such attacks have been purposely implemented.

Figure 3 shows the original Lena, whose dimension is 512×512 pixels, and its watermarked version. The PSNR calculated on these two images is 47.25 db, where the PSNR is defined as $10\log(255^2/MSE)$, being MSE the "mean squared error" computed on Lena and its watermarked version. The user sequence μ is 128 bit long.

Figure 3. “Lena” and its watermarked version.

(a) (b) (c) (d)

Figure 4. Some attacked images of “Lena”.

5.1. Stirmark Attacks

Table 1 reports three main parameters for each attack: the number f of the insertion frequencies used to watermark the image; the *ber*, defined as $(\sum_{i=1}^{f} ber_i)/f$, where ber_i is the number of bit errors reported in the watermark extraction carried out at the frequency γ_i; the PSNR calculated on the watermarked and attacked image. Figure 4 shows some attacked images of Lena: a sharpened image (4(a)), a “median” filtered (factor 9) image (4(b)), an image corrupted by an additive noise (factor 5) (4(c)), a JPEG re-encoded (quality factor 60) image (4(d)). The attacks are named according to the Stirmark definitions.

The results reported in Table 1 show that the proposed procedure can achieve a good performance against attacks that are considered able to prove a high level of robustness without imposing any strong constraint to the length of the codes identifying users. In particular, the *ber* values are always very low and have never prevented the user sequence μ from being correctly re-built from the single sequences $\hat{\mu}_i$ extracted for each test at the different insertion frequencies. In fact, the redundancy assured by the insertion process enables the procedure to behave as other well known watermarking procedures. This also because the procedure allows for choosing the insertion frequencies as well as the regions of the image where to embed the watermark. Moreover, the tests in Table 1 show that the value of f does not result in strongly influencing the final PSNR values. As a consequence, it can be also increased to improve the security level of the procedure without causing a further reduction of the final visual quality of the image.

5.2. Collusion Attacks

For user codes or fingerprints to allow for identifying “colluders” exploiting differently watermarked versions of the same image to produce a new version of the image with no detectable watermark, the watermark embedding method has to be capable of withstanding the collusion attacks [12]. Therefore, some tests based on “linear” and “nonlinear” collusion have been purposely implemented. In the former case, k differently watermarked copies of the same image are linearly combined by averaging the copies with an equal weight to produce a colluded version of the image. In the latter case, an attacked image is created in which each DCT coefficient is the minimum, maximum, and median, respectively, of the corresponding coefficients of k watermarked copies of the same image [12].

Table 2 summarizes the results obtained in the collusion tests. In particular, the codes used in the colluding copies have been generated according to what reported in [2]. Furthermore, in order to correctly assess the procedure behavior independently of the adoption of anticollusion codes, the following conditions have been considered as errors in the watermark extraction:

1. if all the colluding copies present a bit 0 in the i-th position of the embedded codes and a bit 1 is extracted from the colluded copy;

2. if all the colluding copies present a bit 1 in the i-th position of the embedded codes and a bit 0 is extracted from the colluded copy;

3. if the colluding copies present both a bit 0 and a bit 1 in the i-th position of the embedded codes and a bit 0 or a bit 1 is extracted from the colluded copy.

In fact, if the conditions reported above occur in the watermark extraction, the watermarking procedure prevents the anticollusion codes from correctly catching colluders according to the capabilities documented in [2]. Therefore, to reduce the number of errors determined by the above conditions, particularly the condition 3, the expressions reported in (4) have been thus modified:

$$\begin{cases} \hat{W}_{b_m}(\gamma_i) - \hat{W}_{b_n}(\gamma_i) > th \implies \searrow \text{ is extracted} \\ \hat{W}_{b_m}(\gamma_i) - \hat{W}_{b_n}(\gamma_i) < -th \implies \nearrow \text{ is extracted} \end{cases} \quad (5)$$

Attack	f	k	$ber(\%)$
Linear attacks			
Averaging	3	5	1.95
Averaging	9	5	1.89
Averaging	3	10	2.62
Averaging	9	10	2.54
Averaging	3	15	3.97
Averaging	9	15	3.02
Nonlinear attacks			
Minimum	3	5	1.09
Minimum	9	5	0.94
Minimum	3	10	2.1
Minimum	9	10	1.83
Minimum	3	15	3.07
Minimum	9	15	2.84
Median	3	5	1.27
Median	9	5	1.12
Median	3	10	2.35
Median	9	10	2.06
Median	3	15	3.42
Median	9	15	3.08
Maximum	3	5	1.14
Maximum	9	5	1.06
Maximum	3	10	2.28
Maximum	9	10	2.05
Maximum	3	15	3.26
Maximum	9	15	3.03

Table 2. The results of some collusion attacks.

where th is a threshold calculated as

$$th = \omega(JND_{b_m}(\gamma_i) + JND_{b_n}(\gamma_i)), \quad 0.45 < \omega < 0.55$$

and ω is a factor depending on the characteristics of the image. Thus, whenever the expressions in (5) do not allow for extracting an *up* or a *down* symbol from the i-th position of the code retrieved from the colluded copy, it is possible to establish that the colluding copies present both a bit 0 and a bit 1 in the i-th position of the embedded codes, and this enables the anticollusion codes to catch the colluders according to the capabilities described in [2].

The results reported in Table 2 show that the proposed procedure can achieve a good performance also against some relevant collusion attacks. In fact, the *ber* values are low and always allow the embedded codes to identify colluders according to the catching capability documented in [2]. Moreover, the behavior of the procedure results in being rather independent of the number of the insertion frequencies. Finally, the PSNR values have not been reported in Table 2, since they all approximate a value about 42 db.

6. Conclusions

This paper presents a watermarking procedure that directly acts on compressed JPEG images and exploits XML documents to store information needed to the watermark extraction. The redundancy assured by the insertion process enables the procedure to achieve a good performance against the most common and dangerous attacks. Moreover, the procedure robustness can be improved by increasing the number of the insertion frequencies. Finally, the use of XML documents enables the security and robustness levels usually characterizing the nonblind watermarking schemes to be achieved without requiring the unprotected original images to be exchanged in the Internet whenever the watermark extraction has to be performed. In fact, keeping XML documents protected or securely exchanging them in the Internet results nowadays in being almost easy. Furthermore, the XML technology is well supported by the Java world, and this makes the procedure well suited to be adopted in a web context.

References

[1] M. Barni and F. Bartolini. Data hiding for fighting piracy. *IEEE Signal Processing Magazine*, 21(2):28–39, 2004.

[2] D. Boneh and J. Shaw. Collusion-secure fingerprinting for digital data. *IEEE Trans. on Information Theory*, 44(9):1897–1905, 1998.

[3] T.-Y. Chung, M.-S. Hong, et al. Digital watermarking for copyright protection of MPEG2 compressed video. *IEEE Trans. on Consumer Electronics*, 44(3):895–901, 1998.

[4] F. Frattolillo and S. D'Onofrio. Applying web oriented technologies to implement an adaptive spread spectrum watermarking procedure and a flexible DRM platform. In P. Montague et al., editors, *Procs of the 3rd Australasian Information Security Workshop*, volume 44 of *Conferences in Research and Practice in Information Technology*, pages 159–167, Newcastle, Australia, Feb. 2005.

[5] F. Frattolillo and S. D'Onofrio. A video watermarking procedure based on XML documents. In *Procs of the 13th Intl Conf. on Image Analysis and Processing*, LNCS, Italy, 2005.

[6] S. Katzenbeisser. On the design of copyright protection protocols for multimedia distribution using symmetric and public-keywatermarking. In *Procs of the 12th Intl Workshop on Datab. and Expert Systems App.*, pages 815–819, 2001.

[7] S. W. Kim and S. Suthaharan. An entropy masking model for multimedia content watermarking. In *Procs of the 37th Hawaii Intl Conf. on System Sciences*. IEEE CS, 2004.

[8] C. L. Lei, P. L. Yu, et al. An efficient and anonymous buyer-seller watermarking protocol. *IEEE Trans. on Image Processing*, 13(12):1618–1626, 2004.

[9] Y. Wang, J. Ostermann, and Y. Zhang. *Video Processing and Communications*. Prentice Hall, 2002.

[10] A. B. Watson. DCT quantization matrices visually optimized for individual images. In J. P. Allebach and B. E. Rogowitz, editors, *Human Vision, Visual Processing and Digital Display IV*, volume 1913 of *SPIE Procs*, pages 202–216, S. Jose, CA, USA, Feb. 1993.

[11] R. B. Wolfgang et al. Perceptual watermarks for digital images and video. *Procs of the IEEE*, 87(7):1108–1126, 1999.

[12] M. Wu et al. Collusion-resistant fingerprinting for multimedia. *IEEE Signal Processing Magazine*, 21(2):15–27, 2004.

Low Complexity Scrambling Scheme for Compressed Audio Based on Human Auditory Perception Characteristics

Koichi Takagi Shigeyuki Sakazawa Yasuhiro Takishima
KDDI R&D Labs Inc

Abstract

In this paper, we propose a novel scrambling scheme used for secure audio content distribution.

In general, scrambling methods are used for either the baseband domain or the compressed domain. In this paper, we focus our attention on the latter technique to perform the task with low complexity.

Although some conventional methods to alter quantized spectral coefficients have been proposed, they are difficult to execute on mobile terminals due to computational complexity and consumed memory. Therefore, we solve this problem by altering the scalefactor which consumes much less computational resources. Moreover, the proposal also enables control of the audio degradation level by altering with consideration of human perception characteristics.

Computer simulations show that the proposed technique has very low complexity and can provide arbitrarily degraded audio data.

1. Introduction

In recent years, advances in audio compression technology [1]-[3] and the remarkable growth of the Internet have accelerated the distribution of digital audio contents. Meanwhile, the distribution of illegal contents over the Internet continues to increase; hence there is an urgent need to establish content security technologies.

Generally, encryption methods [4]-[6] are employed for digital data management in order to protect content security. In this case, music cannot be played until the file is decrypted.

Meanwhile, there is another cryptographic approach: “scrambling”, which degrades the audio quality, while the scrambled audio contents remain decodable using a normal decoder. Furthermore, the alteration can be reversed. In other words, the descrambling operation recovers original data. Thus, a preview service with degraded quality and minimum understandability is provided maintaining the required security level unlike in the encryption.

Conversely, in the digital contents market, mobile phones offer great promise as a platform for music distribution because they are increasingly prevalent and always carried. A content protection scheme is also required for this platform. Considering the limited memory capacity and CPU power, we need to realize this technique with very little processing resources. Moreover, it is preferable to enable direct scrambling or descrambling of compressed audio data as the contents are distributed in a compressed form.

Several studies have been conducted on scrambling in the compressed domain. Gang [8] proposed flipping the sign or changing the orders of the quantized spectral data with regard to MP3. Herre et. al. [9] proposed using X-OR for the LSB or toggling the sign of the quantized spectral data with regard to AAC.

Since both methods alter quantized spectral data, and they require much more computational time and memory space for descrambling, the conventional methods are not suitable for a mobile terminal. Hence, in this paper, entirely different parameters are investigated for alteration, and relation of the alteration and audio quality are analyzed in view of the human perception model for MP3 audio coding. The analysis derives a control method of the parameter in order to achieve the intended quality.

The following sections are organized as follows. In Section 2, the authors briefly introduce the design of their assumed system and its requirements, and in Section 3, the proposed scrambling method are presented to meet the requirements. This approach is evaluated in Section 4. Finally, conclusions are presented in Section 5.

2. System design and components

The aim of the proposed scrambling system is to enable the proper player to play back original audio data and to be able to control the degraded quality in normal players. In this paper, because we assume the distribution of audio contents to mobile terminals, we can define our premised system requirements as follows:

- Distributed audio data should be in a compressed form such as MP3, AAC, etc.
- Specific playback applications with a descrambling

function to play original audio data should be downloaded only by authorized users.
- A descrambled content shall only exist during its playback.
- The system should allow anyone to be able to download contents freely, even unauthorized users, though the audio quality is degraded.

Under these conditions, the following requirements in addition to the general scrambling/descrambling system (even if the input compressed file is scrambled, it remains decodable using a normal player) are required to realize the proposed scrambling system.

- To save file storage, a single scrambled audio content should be played back at several levels of quality.
- The descrambling process should be realized with very limited computational resources to work on the mobile terminal.
- The audio quality should be controlled by the scrambling intensity on the sender side.
- The audio quality should be controlled by descrambling steps on the receiver side.

3. Proposed method

In this section, the proposed scrambling method for MP3 audio data is presented that satisfies the requirements in the previous section.

3.1. Scrambling

3.1.1. Alteration target

The structure of the targeted audio codec MP3 is shown in Fig. 1. MP3 bitstream consists of 20-30msec frames. Each frame also consists of a) "Header," b) "Side information" and "Main data." And "Main data" consists of c) "Scalefactor" and d) "Quantized spectral coefficients."

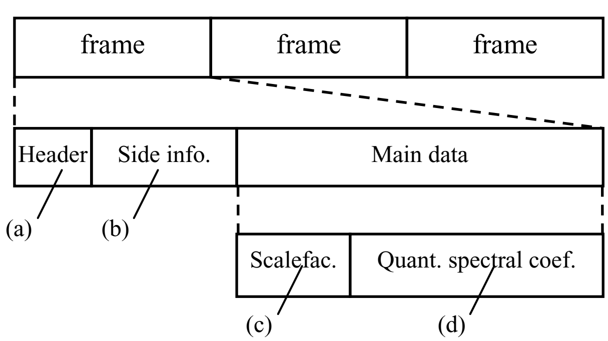


Figure 1. MP3 structure

For altering (d) as in the case of the conventional method, we require an extra 50[KB] or so of memory compared with altering the other parts ((a), (b) or (c)) to expand or store look-up tables for altering quantized spectral coefficients which are Huffman-coded. The extra memory size is too large when compared to a coded frame data size: 0.3[KB] (for 96kbps, mono, 44.1kHz) and a decoded frame size: 2.2[KB]. Moreover, much more time is required for this process. For the reason, it is a major challenge to execute descrambling on a mobile phone if this part is used.

On the contrary, for (a) or (b), the computational cost is very low. However, we can only alter sound pressure levels for all subbands together. Consequently, it is impossible to control audio quality just by altering these values as each frequency band cannot be operated.

Finally, for (c), a scalefactor exists for each scalefactor band and is decoded comparatively easily compared with case (d). This is mainly because the consumed memory size is very small as expanding the Huffman-table is not required.

As a conclusion of the above discussion, (c) the scalefactor value is the optimal variable for scrambling operations

3.1.2. Human perception characteristics

In this section, human perception characteristics are described to explain the proposed method.

Although scalefactor is selected to control audio quality for each scalefactor band, the relation between audio quality and scrambling intensity remains a matter for further study. Thus, a method to control audio quality based on the scrambling intensity is analyzed.

Human auditory perception has the following characteristics [10]:

- Absolute threshold of hearing

The absolute threshold of hearing characterizes the amount of energy required for a pure tone such that it can be detected by a listener in a noiseless environment. The frequency dependence of this threshold has the following characteristics.

  - The audio sensitivity is the highest at around 4 kHz.
  - The lower a sound level (Phon), the weaker the audio sensitivity at low frequency.
  - Audio sensitivity does not increase uniformly with an increase in frequency at high frequency.

- Spectral masking effects

Masking refers to a process where one sound is rendered inaudible due to the presence of another. Simultaneous masking may occur whenever two or more stimuli are simultaneously presented to the auditory system (Fig. 2). Audio coding algorithms are premised on this characteristic, whereby error contributions are masked by a masker.

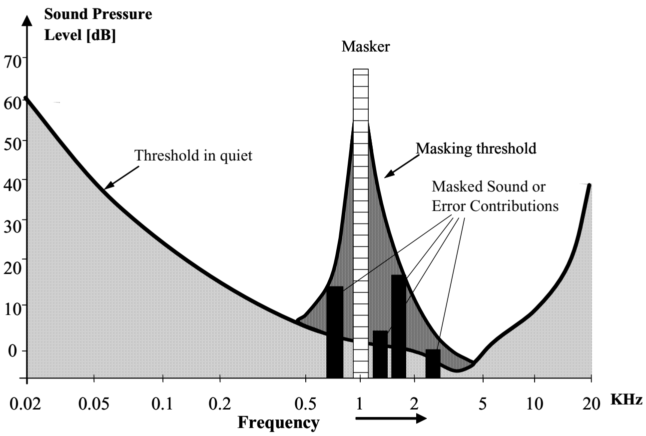


Figure 2. Masking Effects.

In MP3, incrementing a scalefactor value means a 6dB (or 4dB) sound pressure level reduction for the corresponding scalefactor band. In addition, the level reduction reveals the noise signal for the adjacent bands, which were originally masked by the peak level band. This fact implies that audio quality may not change greatly if two adjacent scalefactors are enlarged in the same manner.

From the reason mentioned above, it is necessary to consider human perception characteristics in order to dynamically change subjective audio quality. Therefore, we describe the scrambling procedure taking this into consideration.

3.1.3. Scrambling procedure

The principles of scalefactor alteration are derived from the analysis in Section 3.1.2.

- To operate a spectrum value at the middle frequency (near 4kHz) to degrade audio quality.
- To change the non-adjacent spectrum value.

As a result, we propose the following procedure as the scrambling method:

First, all sub bands are classified into three classes: low, middle and high frequency levels with a central focus on 4kHz. Subsequently, (M is a natural number)

Step 1) One of the scale factors at the middle frequency level is incremented.

Step 2) One of the scale factors at the low and high frequency level is incremented.

…

Step $2M$-1) One of the scale factors that has not been altered at the middle frequency level is incremented.

Step $2M$) One of the scale factors that has not been altered at the low and high frequency level is incremented.

Fig. 3 shows the flow of the above-mentioned scrambling procedure and the descrambling procedure described in Section 3.2. Thus, we can realize the targeted

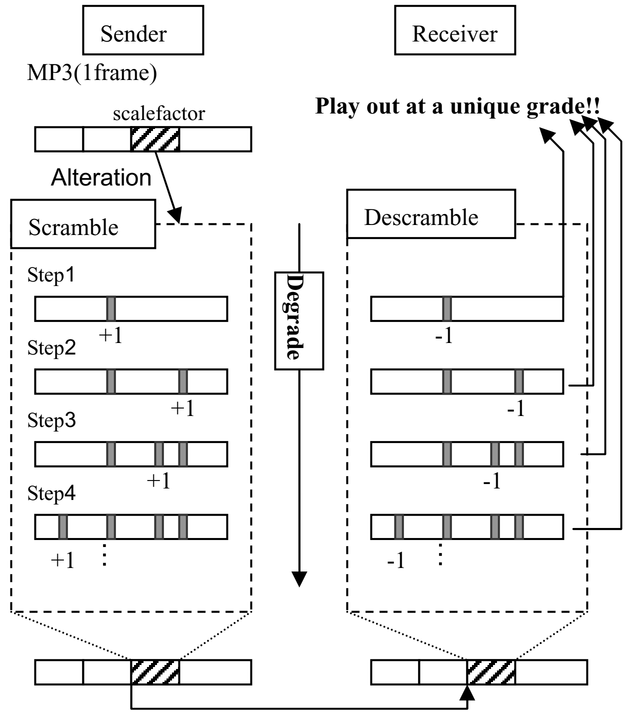


Figure 3. Structure of proposed method.

scrambling system very easily and control the scrambling intensity by the parameter M. These are verified using a computer simulation described in Section 4.

3.2. Descrambling

The procedures for descrambling are as follows:

- Bitstream analysis

  Only the scalefactor part of the bitstream needs to be analyzed.
- Recovering a corresponding scalefactor

  Inverse alteration of the previous section should be performed. This means that the corresponding scalefactor is decremented. The target scalefactors are those altered in Section 3.1.2, and are shared by the sender and receiver side as the secret key in this scramble system.

Furthermore, quality control on the receiver side can also be realized using the following procedures:

1. The sender scrambles audio data at a maximum intensity, and then
2. The receiver descrambles it, but stops in mid stream.

The descrambling process tracks back the scrambling procedure reversely, thus the recovered audio quality is the same as in step$\geq M'$ of scrambling ($M'<M$).

4. Performance Evaluation by Experiments

In this section, some evaluation experiments are conducted to verify the effectiveness of the method described in the previous section.

4.1. Evaluation of descrambling speed

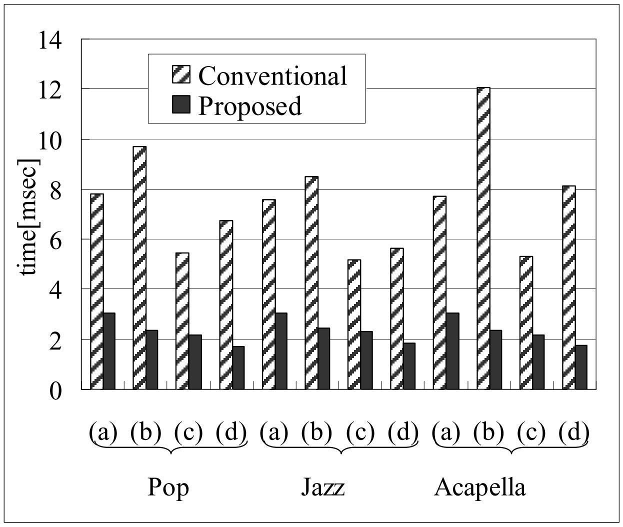


Figure4. Descrambling Speed.
((a)96kbps,32kHz,mono, (b)96kbps, 32kHz,stereo, (c)128kbps,44.1kHz,mono,(d)128kbps,44.1kHz,stereo)

In this section, the processing time for the proposed descrambling method implemented on a PC is measured for relative comparison with a conventional method.

The processing time for descrambling per frame is shown in Fig. 4. By way of comparison, the processing time for reordering spectral coefficients described in [8] is also measured (In this section, this is called the "conventional" method.).

Figure 4 shows that the "conventional" method requires 3-5 times as much processing time as the proposed method. As mentioned above, the "conventional" method requires much more time to expand the Huffman-table for bitstream analysis. It should be noted that these experiments were conducted on a PC, and the performance of the conventional method worsens for cell phones.

Taking the case [128kbps, 44.1kHz, monaural] as an example, the conventional method requires 0.2[sec] to descramble 1[sec] bitstream on PC, but the proposed method only requires about 0.07[sec]. However, even though the processing speed on the PC is about 20 times higher than that of a cell phone, the conventional method on cell phone requires more than 4[sec] (= 0.2 x 20). Because the cell phone generally has a small size cache, the cache missing rate in the process of Huffman decoding increases greatly, resulting in processing speed loss.

4.2. Quality control characteristics

4.2.1. Experiment methodologies

For music data, there are two subjective evaluation methods: one is for high quality audio data[11], and the other is for intermediate quality audio data[12]. In this paper, based on the characteristics of the currently targeted data, the latter, MUSHRA (MUlti Stimulus test with Hidden Reference and Anchor) is used.

Several time-aligned audio signals were presented to the listener who performed on-the-fly switching between these signals using the keyboard and screen. These signals included the original, labeled "Reference," and several anonymous items, arranged in random order. The used data (item) is shown in Table 1.

Table 1. Test Data.

Notation in Section 4.2.2	Description
Orig	Compressed audio data（Original data）
3.5bf/7bf	3.5kHz/7kHz band limited data, respectively. Required as the other references in [12]
Prop[i] <Proposed Method>	The scalefactors of the bitstream are altered as described in Section 3.1.3.: "*i*" shows the step number (*i* = 1,…, 2*M*).
Incr[i] <For Reference>	The scalefactors of the bitstream are altered in ascending order from the lower scalefactor band: "*i*" shows the step number (*i* = 1,…, 2*M*).

The data is prepared for three types of music ((a) Pop, (b) Jazz, (c) Acapella), and two kinds of bit rate/channels (96kbps stereo/64kbs monaural), respectively. The test subjects on this occasion had to grade the basic audio quality of the anonymous items on a scale with five equally sized regions labeled "Excellent," "Good," "Fair," "Poor" and "Bad."

4.2.2. Experimental results and discussion

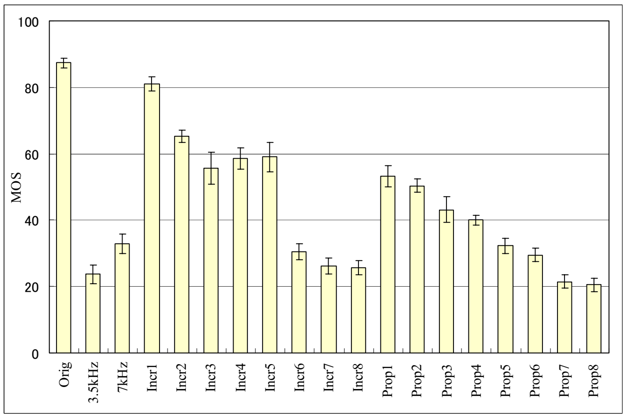
The results for 96kbps stereo contents are shown in Fig. 5. Each graph shows the mean grades (0-100) and 95% confidence intervals. (Because the monaural cases show similar results to the stereo case, only the stereo cases are shown.)

First, we focus on the score movement for "Prop[.]" and "Incr [.]". It can be observed that the evaluation scores for "Prop" decrease monotonically. Meanwhile, the evaluation scores for "Incr" decrease on an irregular basis.

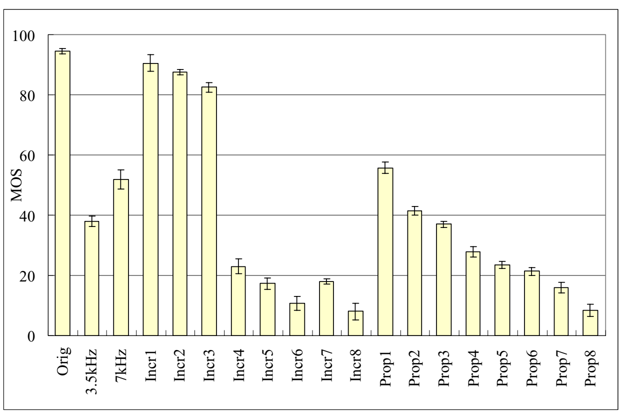
As described in Section 3.1.2, the audio qualities are not uniformly degraded if the scale factors in adjacent sub bands are operated in order such as for "Incr." On the contrary, it is shown that to operate scale factors in separating sub bands allows gradual degradation for each operation. Moreover, a scale factor is observed to operate at a middle frequency sub band, with a significant effect on audio quality. This proves the assumption in Section 3.1.3 to be correct.

It is also observed that the differences among a variety of contents, the number of channels (monaural/stereo) or bit rates do not cause any significant effects in terms of differences in the absolute score. This means the proposed method is generally applicable to any genre of contents.

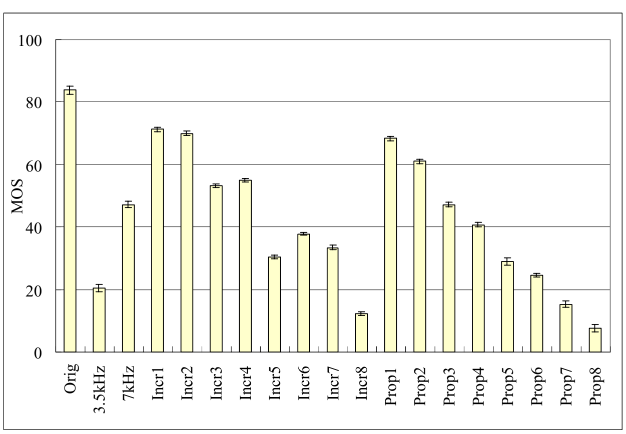
As seen in the results, the audio quality score for "Incr[*i*]" decreases on an irregular basis as "*i*" is incremented. This is attributed to the masking effect described in Section 3.1.2. The spectral analysis in Fig. 6 illustrates this. The left side of Fig. 6 shows the spectral results for reference methods (Incr), and the right side shows those for the proposed methods (Prop). The top left graphs in Fig. 6 show the spectral analysis results for Incr[*i*] (*i*=8,9,10) and the bottom left graphs ((1)-(3)) show their enlargements. The top right graphs in Fig. 6 show those for Prop[*i*] (*i*=9,10,11) and the bottom right((4)-(7)) show their enlargements. An investigation of graphs (1)-(3) reveals that there is a large difference between the spectrum for Incr[8] and Incr[9], while there is just a slight difference between that for Incr[9] and Incr[10]. Moreover, the spectrum for Incr[9] differs more from the original data than that for Incr[10] at around 2kHz, which is shown as the white circles in the figures (graphs (2), (3)). In contrast, there is a steady difference between Prop[9], Prop[10] and Prop[11] as seen in the example of graphs (4)-(7).



(a) Pop, 96kbps stereo



(b) Jazz, 96kbps stereo



(c) Acapella, 96kbps stereo

Figure 5. Performances of degraded (scrambled) audio data

5. Conclusions

A novel audio scrambling algorithm for compressed data is proposed. The most important feature of the method is that it requires low computational resources so that the algorithm can be easily applied to secure music delivery to the cell phone. The second feature is that the algorithm ingeniously utilizes the masking effect and other human auditory perception characteristics, so it achieves gradual degradation control. It should be noted that the evaluation was performed for MP3 audio, nevertheless the principle of the algorithm is applicable to other audio codecs which utilize the human perception model.

6. Acknowledgement

The authors would like to thank Dr. Y. Hirata, Mr. T Asami, Dr. S. Matsumoto, Dr. M. Wada and Mr. Y. Nakajima for their continuous support.

7. Reference

[1] ISO/IEC 11172-3: Coding of moving pictures and associated audio for digital storage media at up to about 1.5Mbits/s, Part 3: Audio, 1993.

[2] ISO/IEC 13318-3: Generic coding of moving pictures and associated audio information, Part 3: Audio, 1998.

[3] ISO/IEC 14496-3: Information technology – Coding of audio-visual objects, Part3: Audio, 2001.

[4] Whitfield Diffie and Martin Hellman, "New Directions

in Cryptography," *IEEE Trans. on Information Theory 22*, p.p.644-654, 1976.

[5] R. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Comm. of the ACM 21*, p.p. 120-126, 1978.

[6] Neil Koblitz, "Introduction to Number Theory and Cryptography, Springer, Second Edition," 1994.

[7] http://www.abanet.org/scitech/ec/isc/dsg-tutorial.html

[8] Litao Gang, "On-line music protection and mp3 compression," *Proc. ISIMP2001*, May 2001, p.p. 13-16.

[9] Herre, J.; Allamanche, E., "Compatible scrambling of compressed audio," *Proc. WASPAA1999*, p.p. 27- 30, Oct.1999.

[10] Ted Painter and Andreas Spanias, "Perceptual coding of digital audio," *Proc IEEE*, p.p.451-513, Apr. 2000.

[11] ITU-R BS.1116-1**:** Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems, Oct. 1997.

[12] ITU-R Rec. BS.1534-1: Method for the subjective assessment of intermediate quality levels of coding systems, Jan. 2003.

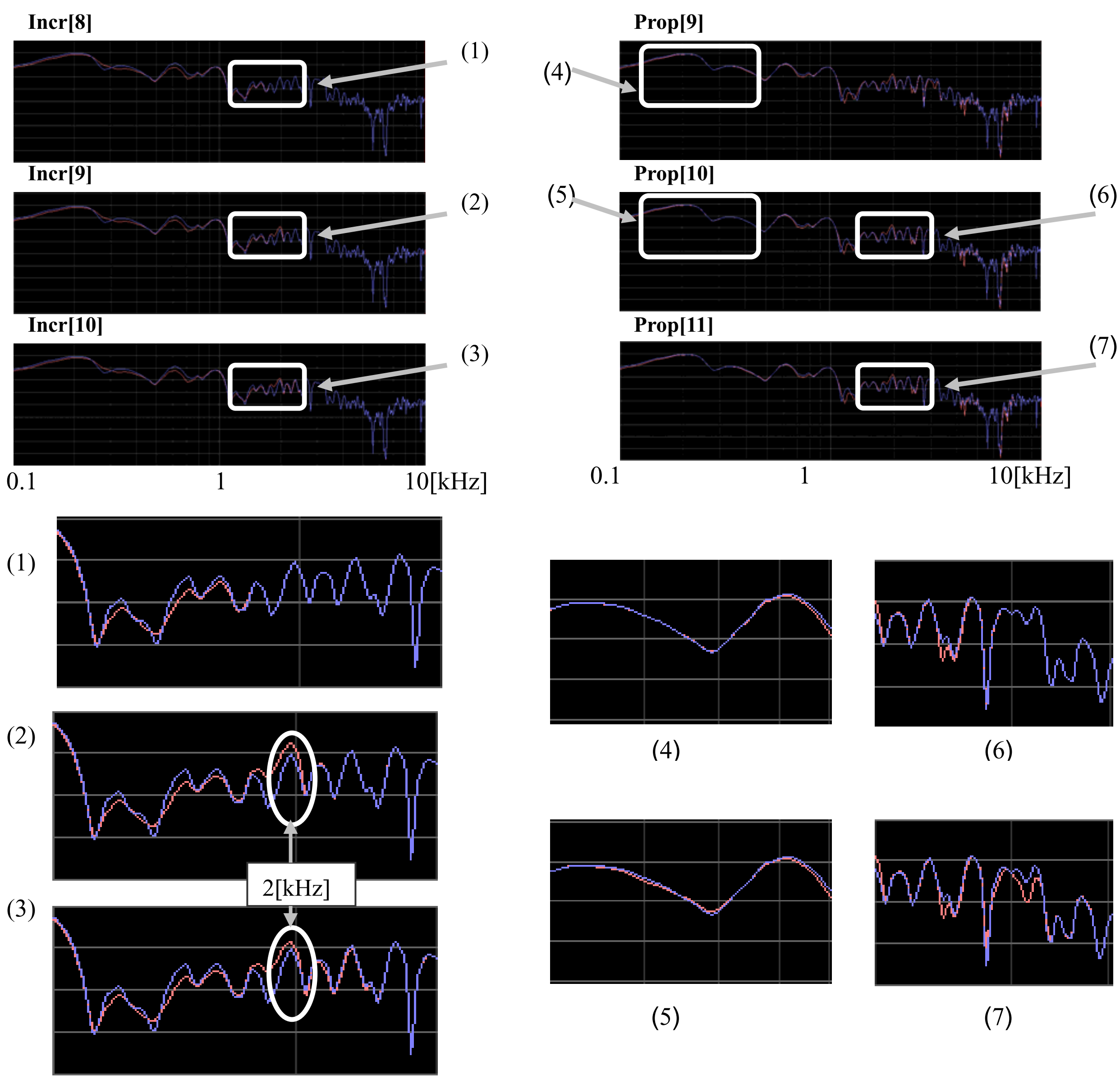


Figure 6. Spectral Analysis Results （blue line: original, red line: scrambled）
Top left: the results for Incr[.]/Top right: the results for Prop[.]/(1)-(7):Their enlargements.

Synchronization in Multimedia Languages for Distributed Systems

A. Guercio[1], A. Bansal[2], T. Arndt[3]

[1]Department of Mathematical Sciences, Kent State University
[2]Department of Computer Science, Kent State University
[3]Department of Computer and Information Science, Cleveland State University

1 Introduction

The rising popularity of multimedia content on the web has led to the development of special-purpose languages for multimedia authoring and presentations. Examples of such languages include SMIL [1], VRML [2], and MPEG4 [3]. These languages support the description of a multimedia presentation containing multiple media sources including both natural and synthetic media as well as media stored in files or streamed live over the network. Some mechanism for specifying the layout of the media on the screen is given as well as a set of primitives for synchronizing the various elements of the presentation. For example, in SMIL we can specify that two video clips be displayed in parallel or that one audio clip be started when another clip finishes. Some of these languages allow for a limited amount of user interactions. A SMIL 2.0 presentation might allow a user to choose a soundtrack in one of several different languages by clicking on a particular area of the presentation. This is accomplished through the incorporation of the events defined in a scripting language such as JavaScript.

While these are well suited for the description of multimedia presentations on the Web, they are of limited use for creating more general distributed multimedia applications since general-purpose programming is only available in the form of scripting languages that have limited power. To support the construction of more large-scale applications approaches such as the use of special multimedia libraries along with a general-purpose language as in the case of Java and JMF [4] or the extension of middleware such as Corba [5] have been proposed. Besides lacking certain essential characteristics for the development of advanced distributed multimedia applications that will be noted below, the use of libraries and/or middleware to achieve synchronization and perform other media related services results in a less well-specified approach than can be achieved by directly extending existing general purpose languages with multimedia constructs with precisely specified semantics. This latter is the approach we follow in our work on multimedia languages that we will describe here.

The language that we want to design should support general-purpose computation; therefore the multimedia constructs whose semantics we will describe should be added to an existing general purpose language such as C, C++ or Java. This is the approach taken by the reactive language Esterel [6]. Reactivity is a very important property for a multimedia language. A reactive system is one which responds to stimuli from the environment [7]. In a multimedia system such stimuli might include user interactions as well as (for example) events generated by the contents of some media stream. The multimedia system must be able to interact with the environment within a short, predefined time period. When used in this context, the difference between reactive systems and interactive systems is that while both may interact with the environment, the latter do not have such a time constraint. The way in which the environment interacts with the multimedia system is through the generation of signals. A signal can be either synchronous (e.g. the reading of some sensing device) or asynchronous (e.g. the recognition of a particular face in a video stream). Our approach to multimedia languages greatly increases the power and flexibility of synchronization by providing synchronization constructs which can be applied not just between media streams, but also between media streams and (synchronous or asynchronous) signals. In fact, in our approach multimedia streams are just a particular type of signal.

The rest of this paper is organized as follows. Section 2 describes the fundamental concepts of signals and streams. Section 3 introduces the various synchronization mechanisms. Section 4 describes the language constructs. Section 5 discusses related research while section 6 describes future research.

2 Signals and Streams

Reactive systems respond to signals generated by the environment. The response must occur within a predefined time period. The signals may have a value. They may also be either periodic or aperiodic. An example of a periodic signal is one which might be generated by a sensor, like a thermometer, which periodically sends the temperature in the form of a signal to the reactive system. An example of an aperiodic signal might be the coordinate values generated by a user using a mouse which are generated only when the user moves the mouse. In order to generalize synchronization of multimedia streams, we define a multimedia stream as a particular type of signal. This allows us to apply the synchronization constructs not just to multimedia streams but to streams and other signals as well.

Reactive multimedia systems often transform or respond to multimedia data which is coming from a sensor such as a digital video camera. The sensor discretizes the continuous media data, converting it into a periodic stream. In such a stream, the multimedia data is associated with a periodic signal. Other types of interactions can produce an aperiodic stream. An example might be a security camera which transmits an image only when motion is detected along a fence line. We model these streams, both periodic and aperiodic as a pair of data and attributes where the data is a sequence of tuples whose elements can be either (elementary) values or tuples. Since multimedia streams are directly associated with a signal, we use the words "stream" and "signal" interchangeably. Formally, we define three types of streams, periodic, continuous and aperiodic, as follows.

Definition 1: A *periodic stream* S^P is a sequence of elements associated with periodic signals. S^P_i is the i-th element in the sequence such that the period *p* does not vary, that is the time between $S^P_{(i+1)}$ and S^P_i is the same as the time between $S^P_{(j+1)}$ and S^P_j, π *i,j* with *iπj*.

Definition 2: A *continuous stream* is the data produced continuously by a sensor. A sensor is used to detect information from the environment. A continuous stream can be modeled as a periodic stream with periodicity $0 < p < \pi$ where π is a small value. The period *p* represents the rate at which the signal is produced.

Definition 3: An *aperiodic signal* can be generated at any time either by an external stimulus or by an event generated after a computation or through a user interaction (such as voice or mouse movement). An *aperiodic stream* S^A is a sequence (possibly of length one) of aperiodic signals. In an aperiodic stream S^A_i represents the i^{th} aperiodic signal.

A multimedia stream is then defined as follows.

Definition 4: A *multimedia stream S* has two components: *attribute-set* and *data*. Three attributes *periodic* or *aperiodic*, number of data elements per unit time, and type of data (such as *audio* or *video* or *music or audiovisual* etc.) are essential. Other attributes are specific to the streams, and vary with different types of multimedia streams.

Let us consider an example of a multimedia stream.

Example 1: A quadraphonic audio stream is a continuous periodic multimedia stream, whose components are:

(i) the *data* represented as a sequence of sampled packets;

(ii) the *attribute-set* A={a_0 = periodic, a_1 = audio, a_2 = 44,100 samples per second, a_3 = no. of channels = 4, a_4 = 16 bits per sample, a_5 = media length, …}.

As an example of an aperiodic stream we have the following.

Example 2: Aperiodic signals have data and attributes as well. An example of an aperiodic signal is *mouse movement*. The components of this aperiodic signal are:

(i) the *data* which describes the mouse movement as a sequence of (X-coordinate, Y-coordinate);

(ii) the *attribute-set* A={a_0 = aperiodic, a_1 ="Mouse Movement", …}.

3 Synchronization

Multimedia synchronization represents some logical relationship (temporal, spatial, spatiotemporal, or logical) between two or more multimedia streams or objects [8]. In the context of research in multimedia computing, however, it is customary to use synchronization to describe only the temporal relationship [9]. Synchronization can also mean *intra-media synchronization*, which defines the temporal constraints within a single multimedia stream; however in general synchronization means *inter-media synchronization*.

Synchronization research can be done on a number of issues [8]. Among these are modeling and specification of synchronization requirements, synchronization algorithms and protocols, and fault recovery in the presence of failures. Our research falls in the first category.

In order for multiple streams to be synchronized, they must share a common clock. In centralized systems it is possible for all streams to use the same physical clock, but this is not possible in distributed systems. The use of multiple physical clocks in such systems is problematic since clock drift can cause skew between the clocks. In order to control this, the multimedia data source insert synchronization points into the streams. The synchronization points can be media points or event counters which preserve the partial ordering of events [10]. In general, inserting more sync points allows for a finer degree of synchronization, but with the cost of added overhead. For synchronization purposes, the multimedia streams need to share a common clock (possibly a logical clock). This common clock provides a common time-base used for synchronization purposes [11].

One of the issues we should take into account while synchronizing media streams is the possibility to have synchronization skew. For example, for good lip synchronization the limit of the synch skew is ±80 msec between audio and video. In general, ±200 msec is still acceptable. For a video with speaker and voice the limit is 120 msec if the video precedes the voice and 20 msec if the voice precedes the video [Li04]. The worst sync skew that can occur depends on how far apart the synch points have been chosen. The closer the synch points, the more resources (such as buffer space) are required when the system is running but more precision in synchronization will be obtained. How should the synch points be chosen if we have several streams with different synch points to be synchronized? Of course, a common synch point for all the streams must be found. If do not need a very strict synchronization or we want to save resources, we explicitly define the synch points to be further apart. For example, if two streams are dependent and one stream has sync points every 2 seconds and the other stream every 3 seconds, a resource saving choice would be to select as the common sync point the least common denominator of the two synch points, i.e. 6 seconds. However, it is highly probable that this choice would not provide great results from the visualization point of view, since it would increase the chance of perceptual distortion. Therefore, this choice is not appropriate in our case where synchronization must be very tight. A more restrictive option is used by selecting the smallest of the values of the synch points of the streams.

3.1 Groups

A distributed multimedia system receives input data from sources, which can be either local or remote. The input data are either streams of media data or asynchronous signals produced by the interaction of the system with the external environment. Because of the temporality of the input data involved in the model, each multimedia stream S generated from the source has an associated clock $c \pi C$ and forms a *multimedia source* Src(S, c). Many streams can be generated at the same instant from different sources. Some of these streams must be treated by the distributed reactive multimedia systems as dependent on one another; other streams must be treated independently. For example, suppose that an audio and a video are produced from two distinct sources and that they must be rendered at the same time. If we later decide to speed up one of the two streams what should happen to the second one? If they are dependent the second should speed up as well, on the other hand if they are independent, the second one should proceed undisturbed. We show the dependence of the streams by defining them to be members of the same group.

Definition 5: A *group G* is defined recursively as:

(i) A single multimedia stream such that the time-base of G is same as the time-base of the stream.

(ii) Two or more multimedia streams sharing a common time base or with their time-bases related through an equation for proper synchronization.

Logically, a group is a tree in which the interior nodes of the tree are groups, and the leaf nodes are multimedia streams. Groups express of dependency of streams and operations performed on the group will influence all of the elements of the group. Operations on a particular multimedia stream of a group (e.g. scaling of the time-base) can either be applied in isolation or to all of the streams in a group to which the stream belongs. We will call the first type of operation *isolated* and the second type *synchronized*.

3.2 The Synchronization Process

The synchronization process consists of a set of spatio-temporal functionalities that enables rendering of multimedia objects in multiple streams to have the same perception as if it happened in real time. *Inter-media synchronization* involving multiple media streams requires the ability to relate the clock of each stream either through a common shared time-base or through an equation relating the two time-bases. If two multimedia streams are grouped together (are dependent) then changing the time-base of one stream similarly affects the time-base of the other stream to maintain synchronization.

Definition 6: Given n streams $S_1, S_2, \ldots, S_n$, in a multimedia group $G = \{S_j, S_{j+1}, \ldots, S_m\}$, and a *synchronization* function f, the application $f(S_i)$ $(1 \pi i \pi n)$ enforces the application of function f on all the multimedia streams $S_1, S_2, \ldots, S_n$ such that the synchronization constraint is maintained.

Consider the lip synching of an audio and a video stream that are played in a lock step manner to give a realistic perception. The two streams are grouped so that time-scaling on either stream (e.g. time stretching or compressing the video) causes the other stream to be time-scaled to maintain synchronization.

The grouping of streams is dynamic and event-driven. That is, certain events may affect a group in one way while another event may affect the group in another way. This dynamic behavior can be obtained either by having multiple orthogonal groups and associating different events with different groups, or by dynamically ungrouping and regrouping the streams through language grouping constructs.

4 Language Constructs

In this section we will describe those language constructs that support synchronization in reactive multimedia systems. Due to space limitations, we are not able to give all of the language constructs. The exact syntax of the constructs will depend on the host language that the multimedia constructs are embedded in.

4.1 Stream Definition

Declarations of media streams are given in terms of the source of the stream (a URL) and the type of the stream (audio, video, audiovisual, etc.) An asynchronous stream can be defined as well. The granularity of the sync points is given as well. Each different type of multimedia stream has a number of attributes whose values are given as part of the declaration (e.g. audio attributes include the name of the stream, number of samples per second, number of channels, number of bits per sample, etc.).

Example 3: The following code is used to declare and initialize an audio stream coming from a remote source. The file name is "speech.mov" and the origin of the media is indicated by the URI address of the source. The audio received can be rendered in the systems at 44,100 samples/sec with 16 bits/sample over 2 channels and the playback rate should not be slower than half of the normal rate, nor more than twice as fast as the normal rate. The stream should have a sync point every 200 milliseconds. The host language is C.

```
audio_stream mm1={source1, "192.168.2.102",
"speech.mov", 44100, 2, 16, 0.5, 2.0, .2}
```

4.2 Grouping Constructs

Media streams are grouped together when dependency between streams needs to be stated explicitly. In particular, grouping clusters one or more media streams or groups for synchronization. The groups are both *dynamic* and *hierarchical*.

Example 4: Suppose we have a video that shows a opera singer singing and we want to add the audio in lip synch mode. To keep up the impression that the soprano is really singing, if we speed up the video, we expect the audio to speed up as well. In order to provide this type of synchronization, we need to indicate that the audio and the video are somehow dependent on each other. Groups are an elegant and efficient way to specify synchronization on multiple streams. The following code groups the two streams.

```
group soprano = {"videostream_opera",
"audiostream_opera"}
```

Groups are hierarchical since a group member can be a previously declared group. They are dynamic since we have commands `ungroup`, to dissolve an existing group, `add_group`, to add elements to a group, `delete_group` to remove elements from a group, and `regroup` to add elements to a new group. Groups have sync points as well. They can either be given explicitly, or computed implicitly as the smallest value of the elements of the group (chosen so as not to lose precision).

4.3 Event Definition

Our constructs make heavy use of events. An event may be generated based on the characteristics of the multimedia streams. For example, the appearance of a particular face in a video stream or a particular voice in an audio stream might cause an event to be generated. The events in turn can affect the synchronization of multiple streams. Events are defined based on the satisfaction of one or more partial conditions. Partial conditions can involve the presence or absence of some signal, the matching of an attribute value or some other condition. Events have a destination (module or object or synchronization process) that they are sent to, as well start and end times and a priority.

Example 5: A user performs a right click of the mouse every time he or she wants to start a video clip; however the video clip cannot be started until the current video clip is terminated and the video must be terminated within a reasonably small range of time (within 5 seconds) otherwise the request is dropped.

```
partial_condition_signal_presence cond1 = {
// Test for signal presence
       NewVideoClip,
// Name of the signal – it is present when
// the video is playing
       yes }  // Test for its presence
partial_condition_signal_presence cond2 = {
       RightClick,
// Present when user right clicks
       yes,
// Test for presence (not absence)
       0, 5 }
// It was present in last 5 seconds
event start_video = { player1,
// Destination of event is renderer
       cond1,
// The partial conditions of the event
       cond2 }
```

4.4 Synchronization

One basic synchronization construct is the loop. It has a number of elements which are played one or more times in sequence. The number of times the loop is repeated can be specified along with a delay time.

Example 6: In the following example, the second video stream is played three seconds after the first. The pair repeats two times.

```
loop {
       times = 2,
       element = video_stream1,
       delay = 3,
       element = video_stream2
};
```

Another important type of synchronization is when we want to play two or more streams concurrently. We support this type of synchronization with the parloop construct. The syntax is similar to the loop construct however the elements of the parloop are played in parallel rather than sequentially. Loops can also be nested as shown in the following example.

```
parloop {
       times = 4,
       element = audio_stream,
       loop {
              times = 2,
              element = video_stream1,
              element = video_stream2
       }
}
```

We can also specify that we want two or more media streams to play in parallel (in other words start at the same time) and end at the same time as well in a parloop construct. In order for this to occur in general one or more of the streams must be stretched (scaled). In order for the scaling to occur, the scaling constraints given as part of the Quality of Service requirements in the declaration of the stream must not be violated.

Loops and parloops are the basic constructs used for synchronization of periodic data streams. They are also the mechanism used to start the playback (rendering) of a multimedia data stream. If we wish to play a stream just once, we use a loop construct with a single element and a single loop time. Note also that the presence of a loop embedded in a program does not cause the execution of the rest of the program to wait until the playback finishes – the execution continues concurrently with the rendering of the media.

4.5 Preemption Constructs

Loops, including infinite loops can be ended prematurely in response to an event. For example, consider a situation where a sequence of advertisements are being displayed on a public terminal, and suddenly a weather warning must interrupt or terminate the current show to provide urgent news. The warning is sent as an asynchronous signal, which may generate an alert event and require showing a text stream, which explains the type of emergency. Other multimedia streams such as an audio signal or a video could follow the text. This situation requires the specification of an abortion of a loop based on the presence of an asynchronous signal (for example the pressing of a button). The advertisement is put inside a loop, and the text media is displayed after the asynchronous signal aborts the loop. There are 2 types of abortion: "strong", "weak" which can be specified in the loop or parloop construct:

Strong abortion performs the immediate interruption at the next multimedia sync point without waiting for the completion of the current cycle of the loop as soon as an aperiodic signal is present.

Weak abortion performs preemption as soon an aperiodic signal is present but will complete the “current” loop cycle that is playing when the signal occurs.

Suppose that we have declared a group consisting of a video stream – MyAdvertisement, a text stream – MyText, and an audio stream – MyAudio. Further assume that we have declared sync points every two seconds for this group. If we want to stop playing MyAdvertisement when an aperiodic signal named StopAdv is present, we can do this as follows:f

```
loop {
       times = 3,
       abort_when = StopAdv,
       abort_type = strong,
       element = MyAdvertisement
}
```

In this example, even though the abort type is strong, we will wait until the next sync point in the media stream to abort the rendering of the stream in order to maintain synchronization with other members of the group. In the worst case, we will wait 2 seconds from the time the signal is present until the abortion occurs.

Suppose the advertisement is shown in sequence with some text. Suppose further that in the presence of the aperiodic signal StopAdv we want to skip the rest of the sequence of ads and text, and skip to the song which is supposed to follow them. However, we don’t want to interrupt an advertisement which has already started. In this case we can use weak abortion as shown below.

```
loop {
       times = 1,
       loop {
              times = 1,
              abort_when = StopAdv,
              abort_type = weak,
              element = MyAdvertisement,
              loop {
                     times = 3,
                     element = MyText
              }
       }
       element = Song
}
```

We can further control the weakness of the abortion and specify other possible synchronization situations by introducing the delayed abort. In this case, the delay value is added to the time required to reach the first synch point after the delay is over. That means that the media stream is rendered for the delay period plus the time to reach the first sync point after the delay is over. Again, let us consider an example.

```
loop {
       times = 1,
       abort_when = StopAdv,
       delay = 3,
       element = MyAdvertisement
}
```

In this example, since MyAdvertisement has sync points every 2 seconds and assuming that strong abortion is the default, the abort will occur between 3 and 5 seconds after presence of the signal StopAdv. If the loop ends before this time, no further delay occurs.

5 Related Research

Athwal [12f] presents a methodology for the synchronization of multimedia streams for engineering and scientific analysis. Since the scientific and engineering phenomena which are recorded and subsequently played back are frequently not well correlated to the human’s visual and cognitive timeframe, it is quite possible that the previously captured data must be played back in a different timeframe – perhaps using slow motion or time lapse or some more complex variability in the playback rate. This is termed time elasticity and it has some relationship to our notion of stretching grouped multimedia streams. It should be noted that the methods described in [12] are suitable only for prerecorded multimedia streams and that much of the methodology is concerned with minimizing resource usage during the synchronized play back of such streams.

Besides the limitations as far as application area and type of multimedia streams, the focus of that work is different from our research in that the focus is on implementation details for systems for synchronization rather than on languages to allow for the expression of synchronization. The work also differs in that it lacks the notion of sync points, which can be defined for our groups and loops, which allow a high level of flexibility for synchronizing multimedia streams under many different quality of service requirements. Furthermore, interaction of multimedia streams with aperiodic signals is not even considered. The same points about the difference of focus of our work and this one could be made about most of the other recent research on synchronization.

Cameron [13] proposes a model for reactivity in multimedia systems; however their notion of reactivity is much different than ours. They discuss multimedia systems in which a multimedia artifact (i.e. a multimedia stream) can react to discrete *events*, such as an audio player reaching the end of a track as well as to continuously evolving behaviors such as the volume of an audio track. They make a distinction between a series of discrete events and a continuously evolving behavior. Since behavior is an author-level abstraction, which therefore hides implementation details of the media streams, the approach is more suited for use as a high-level authoring tool for multimedia presentations rather than for construction of distributed multimedia systems.

A number of XML-based multimedia languages have been proposed lately. None of them provide all of the capabilities described in this paper, although some provide complementary capabilities. For example, Gu [14] introduces HQML, an XML-based language to express the quality of service requirements of distributed multimedia systems. Another example is the multi-modal presentation markup language (MPML) [15] which is an easy to use XML-based language enabling authors to script web-based interaction scenarios featuring life-like animated characters.

6 Conclusions and Future Research

A model for distributed multimedia systems which incorporates the synchronization constructs discussed in this paper along with an active repository which allows for the constant testing of the multimedia data for deterministic and non-deterministic events is given in [16]. The behavioral semantics of the language constructs have been developed as well in order to provide a formalism for verification, compilation, and validation. The semantics incorporates the temporality and the communication aspects of the system and uses a variation of the π-calculus [17] for modeling distributed reactive multimedia systems. The π-calculus has its roots in the CCS (Calculus of Communicating Systems) [18, 19, 20], which is able to describe interactive concurrent systems as well as traditional computation. The π-calculus adds to the CCS, mobility of the participating processes and uses the transmission of processes as values, and the representation of data structures as processes. This research will be presented in a future paper.

References

1. *Synchronized Multimedia Integration Language 2.0 Specification*, http://www.w3.org/TR/smil20/ Aug. 2001.
2. *Virtual Reality Modeling Language*, ISO/IEC14772, 1997 http://www.web3d.org/x3d/specifications/vrml/ISO_IEC_14772-All/part1/concepts.html.
3. H. Kalva, L. Cheok, A. Eleftheriadis, "MPEG-4 Systems and Applications", *Proc. of the 7th ACM Intl. Conf. on Multimedia (Part 2)*, Orlando, Florida, pp. 192-192, October 1999.
4. R. Gordon, S. Talley, "*Essential JMF – Java Media Framework*", Prentice Hall, 1999.
5. Object Management Group, "*Control and management of A/V streams specification*", OMG Doc. telecom/97-05-07, 1997.
6. G. Berry, G. Gonthier, "The ESTEREL Synchronous Programming Language: Design, Semantics, Implementation," *Sci. of Comp. Progr.* vol. 19, no. 2, pp. 87-152, Nov. 1992.
7. G. Berry, "The Foundations of Esterel", in *Proof, Language and Interaction: Essays in Honour of Robin Milner*, G. Plotkin, C. Stirling and M. Tofte ed., MIT Press, pp. 425-454, June 2000.
8. N.D. Georganas, R. Steinmetz, T. Nakagawa, "Guest Editorial on Synchronization Issues in Multimedia Communication", *IEEE J. on Sel. Areas in Comm.* vol. 14, no. 1, pp. 1-4, 1996.
9. L. Li, A. Karmouch, N.D. Georganas, "Multimedia Teleorchestra with Independent Sources: Part 1&2 – Temporal Modeling of Collaborative Multimedia Scenarios", *ACM/Springer Multimedia Sys.* vol. 1, no. 4, pp. 143-153, 1994.
10. L. Lamport, "Time, Clocks and the Ordering of Events in a Distributed System", *Comm. of the ACM*, vol. 21, no. 7, pp. 558-564, 1978.
11. R. Gordon, S. Talley, "*Essential JMF – Java Media Framework*", Prentice Hall, 1999.
12. C.S. Athwal, J. Robinson, "Synchronized Multimedia for Engineering and Scientific Analysis", *Multimedia Systems,* vol. 9, pp. 365-377, 2003.
13. H. Cameron, P. King, S. Thompson, "Modeling Reactive Multimedia: Events and Behaviors", *Multimedia Tools and Applications*, vol. 19, pp. 53-77, 2003.
14. X. Gu, K. Nahrstedt, W. Yuan, D. Wichadakul, "An XML-based Quality of Service Enabling Language for the Web", *J. of Visual Lang. and Comp.* vol.13, no. 1, pp. 61-95, 2002.
15. H. Prendinger, S. Descamps, M. Ishizuka, "MPML: a Markup Language for Controlling the Behavior of Life-like Characters", *J. of Visual Lang. and Comp.* vol. 15, pp. 183-203, 2004.
16. A. Guercio, A.K. Bansal, "TANDEM - Transmitting Asynchronous Non Deterministic and Deterministic Events in Multimedia Systems over the Internet", *Proc, of DMS 2004*, pp. 57-62.
17. R. Milner, "*Communicating and Mobile Systems; the π-calculus"*, Cambridge University Press, 1999.
18. R. Milner, "A Calculus of Communicating Systems", *LNCS*, vol. 92, Springer-Verlag, 1980.
19. R. Milner, "*Communication and Concurrency*", Prentice Hall, 1989.
20. Y. Wang, "CSS + Time = an Interleaving Model for Real Time", *LNCS* 510, pp. 217-228, 1991.

Uncertain topological relations for mobile point objects in terrain

Karin Silvervarg, Erland Jungert
Swedish Defence Research Agency
karin@foi.se, jungert@foi.se

ABSTRACT

This paper proposes a simplified solution to how uncertainties in relations can be handled when concerned with relatively small, mobile objects, for instance vehicles, in a spatial temporal database system. Incompleteness, inconsistency, vagueness, imprecision, and error both in the sensor data and from the processing of that data result in uncertainties. These uncertainties can be handled by assigning broad boundaries to the objects. There are models for how to handle broad boundaries of spatial object in the general case, but in this special case a lot of those relations are not applicable. Thus a simplified way to manage topological relations with respect to moving artifacts in the terrain is proposed. A solution for how to treat relative positioning between these objects is also proposed.

1. INTRODUCTION

Sensor data are gradually used more as input to a large variety of systems. Examples of such systems are command and control systems for both military and civilian purposes. In the latter case the applications focus may be concerned with emergency management systems. Other examples are robotics, safety and security systems. In many of these systems query languages are required. When designing query languages for sensor data sources a number of problems occur. These problems relates in part to the types of sensors used but also to those cases when multiple sensors are used to collect data more or less simultaneously. In the latter case methods for sensor data fusion will be required. As a consequence, query languages for multiple sensor data sources must be designed to fuse sensor data. The fusion techniques may be of different types. An example of a query language with an integrated sensor data facility is discussed in [6]. A particular problem that occurs in conjunction with the use of sensor data is the uncertainty in the data. The data uncertainties can be of different types, e.g. incompleteness, inconsistency, vagueness, imprecision. Basically, uncertainties in sensor data are generally due to imperfections in the sensors and in the sensor data analysis. These uncertainties must be handled properly by the query language. However, the way the uncertainties should be handled depends to a large extent on the class or classes of problems the queries should be applied to. The uncertainties have, in particular, effects on the queries applied to the spatial relations that occur between extended geographical object [3] or on spatial relations between man-made objects or man-made objects and extended objects where the latter can be part of the background context. A general approach to handle the uncertainties in these problems is to represent the object with what sometimes is called *a broad boundary.* The object thus has an uncertainty component that depends on extension and position. This representation includes an interior of the objects that definitively belong to the objects. The broad boundaries, on the other hand, include parts that are part of the object and parts that are not. Consequently, the broad boundary mirrors the uncertainties of the object. This way of representing the uncertainties are in many ways related to the theory of rough set [8]**.** Since the degree of uncertainty in sensor data primarily depends on the type of sensor this means that different sensors under different circumstance deliver uncertainties of different character, which has consequences on how the query language should handle the actual uncertainties. The solution to this problem will be further discussed subsequently in this paper for small man-made objects in relation to larger objects that are part of the background. The background is here sometimes called the context.

The general structure of this work can be described as follows. Chapter 2 introduces the objectives. Chapter 3 gives a brief overview of the query-system that is used. Chapters 4, 5 and 6 describe different issues concerning topological relations. Chapter 7 describes other relations that can be applied to mobile point objects. This is followed by the conclusions and directions for future work in chapter 8.

2. OBJECTIVES

The main objective of the work discussed in this here is to, by default; consider existing uncertainties in input data. The input data used in the query language discussed here are basically coming from various kinds of sensors. The sensors are generally of image generating types, such as IR, and laser-radar. The uncertainties in these sensors are for the most part due to imperfections in the sensors. As a consequence, when a query is applied to multiple sensors the uncertainties of each sensor must be taken into account when the query answers are determined and before the answers are delivered to the users. In earlier work the uncertainties were just concerned with how certain the answer is with respect to the requested object type(s). This is by no means sufficient since the sensor data uncertainties will have consequences on other aspects of the queries as well. Among those aspects can uncertainty in position and speed of various entities be mentioned; where these aspects generally are called status variables as they quite often are subject to changes contrary to ordinary attributes. The focus of this work is to consider how these uncertainties affect a number of relations that will occur between the entities. These relations are generally of spatial type that concern man-made and background objects.

3. ΣQL

The query language that we use is called ΣQL [1]. The system [7] is divided into a visual user interface, a query processor and the sensor nodes to which the sensor data sources are attached. The query processor includes a knowledge system that operates in conjunction with an ontology. The purpose of this knowledge system is to support automatic selection of the sensors and sensor data algorithms for the data analysis that together deliver the information used to respond to a query. In a first set-up of the query processor the actual sensors have been a digital-camera, an infrared camera and a laser radar. However, the system is not limited to these three sensor types; others can, on demand, be attached as well. The sensor nodes include also means for target recognition. For this purpose a database containing a library of target models is attached to the ΣQL-processor. The target models stored in this library are used by the image analysis processes to recognize objects found in the sensor data inquired by the users. A meta-database containing the available information that has been registered by the sensors is also attached to the query processor. The query system includes, contrary to conventional query languages, a sensor data fusion module. The purpose of this module is primarily to fuse information extracted from the sensor data, which correspond to sub query results. These data emanate generally from multiple sensors whose sensor data altogether are of heterogeneous type.

The visual user interface is designed to allow both simple and complex queries of spatio/temporal type. The user indicates the area of interest (AOI) in a map and the actual time interval of the query and finally, through a part of the underlying ontology, determines the requested object type or types. In this approach of the query language basically just vehicles are requested. These vehicles can be selected through the visualization of the ontology.

For the time being, complex queries concern vehicles and different types of spatial and temporal relationships that may occur between the vehicles and between vehicles and background information [9]. Background information generally means geoinformation. The most important spatial relationships are *topological relations*, *directions* and *distances*. Queries that allow the combination of spatial and temporal conditions are possible as well. The queried objects are found in sensor data and include sets of attribute and status values. Associated with the object information is also a belief value, which indicates how much the system believes that a certain object is of a specific type. Thus a belief value is a quantification of the uncertainty associated with the sensors and the sensor data they produce. Consequently, the belief values serve a purpose but are by no means useful when determining which relations that may exist between the objects. Due to the uncertainties in the sensor data, these relations include uncertainties, which must be considered when the system responds to the queries. The expressions used are normally found in where-clauses in traditionally text oriented query languages contrary to the situation here where the interface is purely visual.

4. TOPOLOGICAL RELATIONS

Topological relations concerns the way in which two geometrical objects can relate to each other in two dimensions. In [4], Egenhofer identifies eight atomic topological relations for areas *disjoint contains, inside, equals, meet, covers*, *coveredBy*, and *overlap*, see figure 1. These relations assume that the exact sizes of the surfaces are known.

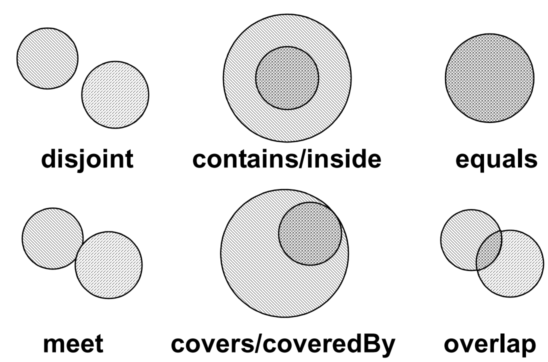


Figure 1. The eight topological relations. Contain/inside and covers/coveredBy are respectively two cases since it depends on which area is surrounding which.

When uncertainties are introduced these relations are not sufficient since the exact positions of the edges of the surfaces are not known. One way to handle this is to add a broad boundary [2]. Then the center of the area signifies the absolutely certain minimum area and the outer boundary signifies for instance the 80% certainty for the area. The resulting topological relations of the areas with broad boundaries are not eight but 44 or 52 [3] if we allow holes in the areas. However, in the work described here this number has been considerably reduced due the object types that are subject to the study, that is in this case some obvious simplifications have been made, which will be discussed further subsequently.

5. TOPOLOGICAL RELATIONS BETWEEN POINT OBJECTS

In our system the focus is on mobile objects acquired from sensor data. Mobile objects can have uncertainties concerning both the size of the objects and their locations. These uncertainties could be handled with broad boundaries. A consequence of limiting ourselves to mobile objects is that the uncertainties in size are limited, as well. If we are able to identify the type of the objects, for example, car or truck, then the uncertainty in size becomes even more limited. The magnitude of the uncertainty in size is negligible compared to the possible uncertainty in position. Even the size of the objects can be neglected compared to the usual size of the uncertainty in position. Consequently, we can approximate the mobile object with only an area that equals the uncertainty in position. The topological relations between two areas are limited for the eight basic relations described in figure 1, but we should consider that since we are taking about mobile objects they cannot in practice overlap, not even a little. Thus the relations *contains, inside, equals, covers*, *coveredBy*, and *overlap* have no equivalence in reality for this type of objects and may for this reason be excluded. Mobile objects may only by close to each other. This is called the proximity, which is equivalent to the topological relations *meet, contains, inside, equals, covers*, *coveredBy*, and *overlap,* or they can be distant which corresponds to the relation *disjoint*. Thus these eight relations can be reduced to just two, see figure 2.

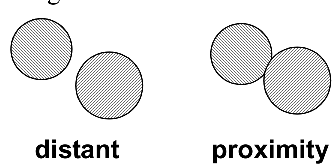


Figure 2. The only two relevant topological relations when concerned with mobile artifacts related to mobile artifacts.

6. TOPOLOGICAL RELATIONS BETWEEN POINT AND BACKGROUND OBJECTS

As was described in chapter 5 mobile objects with uncertain positions can be approximated with an area corresponding to the uncertainty in the position. Background objects, for instance forests, lakes, cities, on the other hand are better approximated with an area including a broad boundary. The kernel of the area corresponds to the part of that with certainty belongs to the object and the broad boundary corresponds to the uncertainty part. As described in section 4, the number of possible topological relations between two simple areas is eight according to Egenhofer [4]. If the areas have broad boundaries the number of possible relations is 44 according to Clementini et al [2].

Another aspect to take into consideration is that since the boundaries of the areas are based on uncertainty it does not seem to be realistic to include a relation that requires the borders to coincide exactly, like the classical relation *meet*, see figure 1.

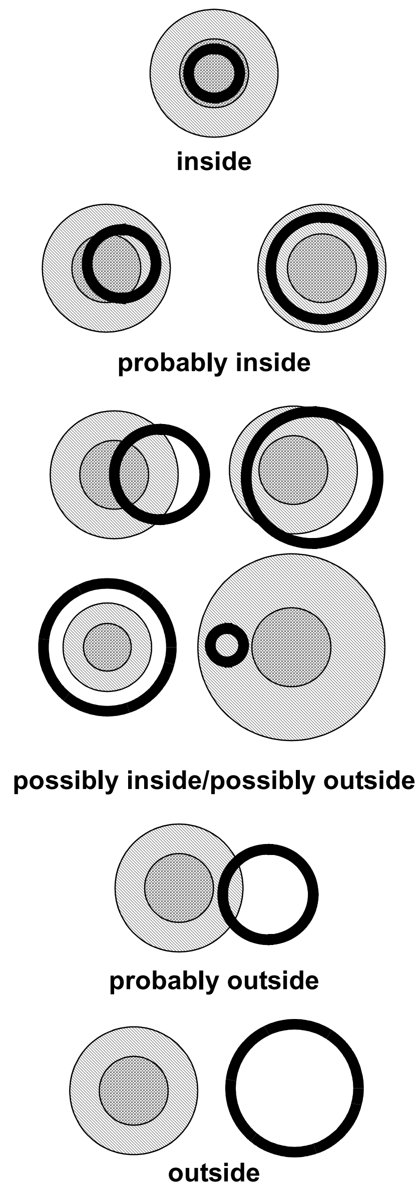


Figure 3. The nine topological relations between a simple area and an area with a broad boundary when the areas do not contain any holes.

The possible topological relations between an object with a simple area and an object with an area with a broad boundary can be expected to be a subset of the 44 relations given some limitations. These limitations can be formulated by two simple rules:

1. The point object is a simple area, while the background object has an area surrounded by a broad boundary.

2. The borders of the objects never coincide exactly.

When reviewing the 44 relations and removing all variants that break at least one of these two rules the results is the nine topological variations. Since the simple area is only an approximation of a point object it can never *cover* or *overlap*, etc. Thus the relations have been grouped into five groups; *inside*, *probably inside*, *possibly inside/possibly outside*, *probably outside* and *outside*, see figure 3. In effect the result is only 4 possible relations.

In [3] Clementini et al introduced relations between two area objects with broad boundaries that may have holes as well. This gives an additional eight relations. When evaluating those relations we add one rule to our set:

3. The point object may not have a hole in its area.

The review of these eight relations result in three remaining relations that are applicable to our case. All three of these relations can be put into the existing group *possibly inside/possibly outside*, see figure 4.

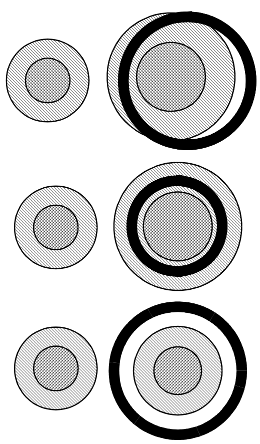


Figure 4. The three additional topological relations between a simple area and an area with a broad boundary containing holes.

7. OTHER POINT OBJECT RELATIONS

Determination of the position of an object based on one or more sensor data sources will always be associated with some types of uncertainty, thus we can never be sure that the given coordinate values are correct. For this reason, a measure of the uncertainty must be available. Usually, for each given position the area of uncertainty is normally represented with an ellipse that sometimes is generalized into a circle. The size and shape of this area depends generally on the sensor type. However, in a query language, which has to respond quickly to the various queries such uncertainty areas are unpractical. For this reason the areas of uncertainty must be replaced with something more efficient. In this work, we have chosen to describe the location uncertainty of a point object by a rhombus. This is motivated not just by its usefulness as a descriptive structure of the position uncertainty but also because it is a convenient way to determine object relations of the types that are discussed in this work. Furthermore, it is also useful, for the determination of directions between a pair of objects, where the secondary rhombus can be found in relationship to the primary one, as the area where the direction cannot be sufficiently correct settled may be vary depending on the uncertain location. The use of such a structure for determination of directions is quite common and variations of it have been used during a long time. Figure 5 shows the orientation of the rhombus and its directional structure relative to a local coordinate system. In 5a the directions are global (north, ...) while in figure 5b the direction are just local (in_front_of, ...) to the object. The area covered by the rhombus is called the *proximity* of the object. Observe also, for instance, that here the direction of a point between *north* and *northeast* is described by the interval [*north, northeast*] or alternatively if the interval is open as]*north, northeast*[. Again this way of describing direction is due to the presence of uncertainties. Thus it cannot be said that the direction of an object is just north, no matter if the given coordinates says so. Instead, again because of the uncertainty of the position, it is]*northwest, northeast*[. Consequently. if the direction of an object relative another one should be determined then the corresponding directional interval should be determined. However, if the object or at least a part of it falls inside the proximity then the direction cannot be determined and for the distance we can just say that the distance between the two objects are very close. This is simply expressed by saying that the objects are in the proximity of each other.

The area of proximity is described by its four corner points by means of its local coordinate system:

$(\varepsilon,0)(0,\varepsilon)(-\varepsilon,0)(0,-\varepsilon)$

It is also easy to see that

$\varepsilon = |\Delta x| + |\Delta y|$

determines the edge of the proximity area.

Here ε is depends on the maximum positional error, i.e.:

$$maxerror = \varepsilon/(\sqrt{2})$$

A consequence of this is that the rhombus is somewhat larger than the actual area of uncertainty but this is negligible since the difference does not contribute much to the positional error. After all, other types of uncertainty areas are approximations as well.

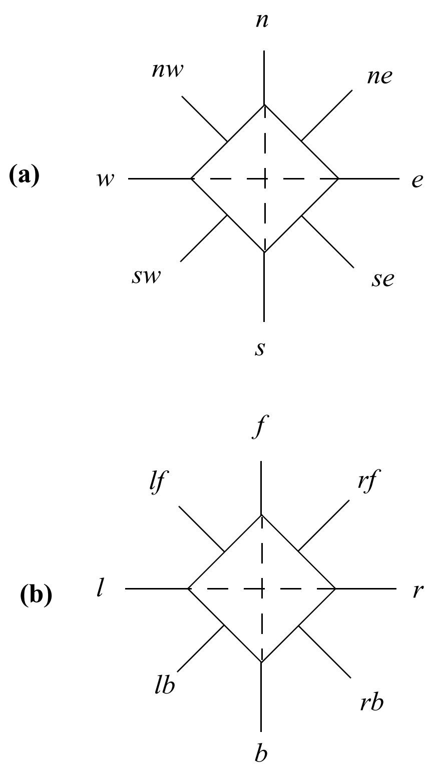


Figure 5. The rhombus proximity representing an object location and its area of uncertainty including also the directions outside that area for both global and local directions.

Determination of the direction of an object relative to another object, both with uncertain positions, is quite trivial and does not require any heavy calculations and can be illustrated by the following cases in terms of rules. An object in the open interval between north and north-east is delimited by the rhombus, the y-axis and the ne-line, i.e.:

If $|\Delta x| + |\Delta y| > \varepsilon$ and $|\Delta x| > 0$ and $|\Delta y| > |\Delta x|$ then] *n, ne* [

If instead the interval is closed we get

If $|\Delta x| + |\Delta y| >= \varepsilon$ and $|\Delta x| > 0$ and $|\Delta y| >= |\Delta x|$ then [*n, ne*]

For an object with the direction *north of* the result is delimited with the edge of the rhombus and the nw and the ne lines, i.e.

If $|\Delta x| + |\Delta y| > \varepsilon$ and $|\Delta y| > |\Delta x|$ and then] *nw, ne* [

A final illustration is an object inside or on the edge of the proximity which is described by

If $|\Delta x| + |\Delta y| <= \varepsilon$ then *proximity*

The methods for determination of the relations between two objects with uncertain positions can now be introduced. This can be illustrated by the two cases in figure 6. The case to the left in the figure shows two objects, A and B, where the areas of uncertainty are overlapping. The conclusion of this is that object B is in the proximity of A. In the right alternative there is no overlap, which indicates that B is outside A and that the direction of B relative A is [*n, e*]. Clearly, the inverse relations are in both case equivalents.

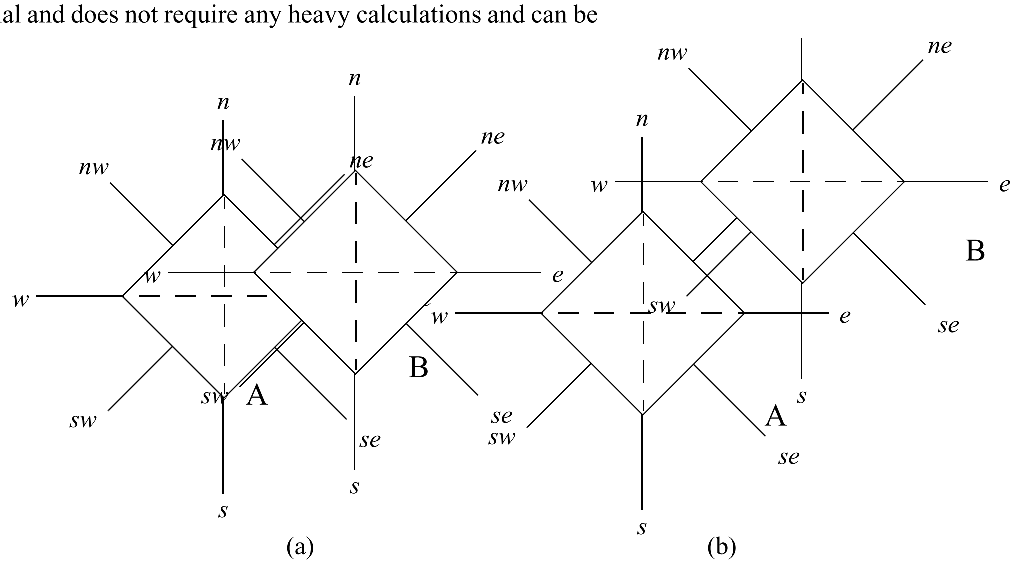


Figure 6. Some possible relations between the objects A and B; (a) B is in the proximity of A; (b) B is outside the proximity of A and B is [*n, e*] of A.

This methodology can be used for determination of qualitative distances as well. The set of qualitative distance measures proposed here is {*proximity*, *close*, *distant*}. This set can, of course, be extended but for the time being it is sufficiently adequate. The qualitative distance structure is illustrated in figure 7. The close and distant distances can be determined from the following rules:

If $|\Delta x| + |\Delta y| >= \varepsilon$ and $|\Delta x| + |\Delta y| < K\varepsilon$ then *close*

If $|\Delta x| + |\Delta y| > K\varepsilon$ then *distant*

where K is a constant that is application dependent. *Proximity* is determined from the rule introduced earlier in this section.

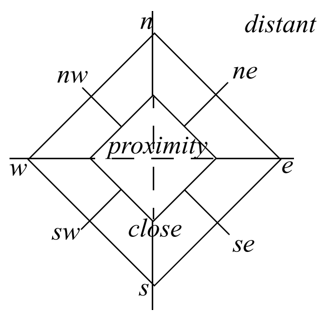


Figure 7. The qualitative distance structure for point objects.

8. CONCLUSIONS

In this paper a solution for how to handle mobile objects detected by sensors has been presented. The focus has concerned with how to handle uncertainties caused by the sensors that affect the reasoning about those objects. Firstly, mobile objects can be approximated with an area equivalent just to the uncertainty in position. A consequence of this is that management of mobile artifacts with uncertain positions differ from how areas with uncertain boundaries should be treated. Finally a way of reasoning qualitatively about relations between such objects has been proposed.

In the paper the uncertainty areas of the point objects are approximated with a circle or a rhombus. In many cases that is a good approximation, but for some sensors that is not a sufficiently good approximation. Future work will include looking into how this work can be generalized to include sensors with more irregular uncertainty areas as well.

Another research topic that has not yet been explored is how to handle uncertainties over time. Apart from affecting relations like before and after it might also influence the relative position if the two objects were not detected at the same time.

9. REFERENCES

1. Chang, S.-K., Costagliola, G., Jungert, E and F. Orciuoli, *Querying Distributed Multimedia Databases Data Sources in Information Fusion Applications*, IEEE Trans. on Multimedia, Vol. 6, No. 5, October 2004, 687-702.
2. Clementini, E. and Di Felice, P., *Approximate Topological Relations*, International Journal of Approximate Reasoning, v 16, n 2, February, 1997, p 173-204.
3. Clementini, E. and Di Felice, P., *A spatial model for complex objects with a broad boundary supporting queries on uncertain data*, Data and Knowledge Engineering, vol. 37, pp. 285-305, 2001.
4. Egenhofer, M., *Deriving the combination of binary topological relations,* Journal of Visual languages and Computing, Vol 5, pp 133-49.
5. Hall, D. L. and Llinas, J. (Eds.), *Handbook of Multisensor Data Fusion*, CRC Press, New York, 2001.
6. Horney,T. , Jungert, E., Folkesson, M., *An Ontology Controlled Data Fusion Process for Query Language,* Proceedings of the International Conference on Information Fusion 2003 (Fusion'03), Cairns, Australia, July 8-11.
7. Horney, T., Ahlberg, J., Jungert, E., Folkesson, M., Silvervarg, K., Lantz, F., Franssson, J., Grönwall, C., Klasén, L., Ulvklo, M., *An Information Sustem for target recognition*, Proceedings of the SPIE conference on defense and security, Orlando, Florida, April 12-16, 2004.
8. Pawlak, Z., *Rough sets*, International Journal of Computer and Information Sciences, 11(5):341-356, 1982.
9. Silvervarg, K. and Jungert, E., *Visual specification of spatial/temporal queries in a sensor data independent information system*, Proceedings of the tenth International Conference on Distributed Multimedia Systems, San Francisco, California, September 8-10, 2004, pp 263-268.
10. Worboys, M. F., *GIS - A Computing Perspective*, Taylor & Francis, london, 1995, pp 99-100.

Visual Languages for Non Expert Instructional Designers: A Usability Study

Gennaro Costagliola, Andrea De Lucia, Filomena Ferrucci, and Giuseppe Scanniello
Dipartimento di Matematica e Informatica, University of Salerno,
Via Ponte Don Melillo, 84084 Fisciano (SA), ITALY
E-mail: {gcostagliola, adelucia, fferrucci, gscanniello}@unisa.it

Abstract

In this paper we propose a usability study on a visual language based tool for the definition of adaptive learning processes. To this aim, we recruited a group of seven teachers with different teaching experiences to evaluate the usability of the proposed tool and as well as to assess the value of visual languages in the definition of learning processes. Interesting results in terms of satisfaction and performance have been achieved on the recruited teachers without computer science background.

1. Introduction

The current trend of academic and industrial realities is to increase the use of e-learning. In particular, software tools supporting the critical task of instruction design should provide automated support for the analysis, design, documentation, implementation, and deployment of instruction via the Web. On the other hand the generated knowledge contents should aim at encouraging high self-expectations, satisfaction, and welfare by respecting the diversities among students. Adaptive learning processes represent a possible solution to improve these significant aspects.

To this aim in [4] an approach based on a visual language hierarchy has been proposed to define learning processes based on synchronous and asynchronous e-learning activities. The adaptive knowledge content is one of the most meaningful asynchronous e-learning activities and is defined by the Adaptive Self consistent Learning Object SET (ASCLO-S) language. ASCLO-S is a special case of flow diagrams where instructional designers define classes of users, represented by stereotypes, and specify for each of them the more suited adaptive learning process.

In this paper we propose a usability study on the ASCLO-S editor, which is turned out to define adaptive learning processes. This study aims on one side at evaluating the usability of the tool and on the other side at assessing the value of the visual languages in the definition of teaching environments. To this end a group of heterogeneous teachers has been recruited. The group was composed of seven teachers with different know-how and teaching experiences.

The remainder of this paper is organised as follows. Section 2 presents related work, while the ASCLO-S language and editor are highlighted in Section 3. The usability test and the achieved results are illustrated in Section 4 and Section 5, respectively. Section 6 discusses the achieved results and compares the ASCLO-S editor with a commercial authoring tool. Final remarks and future work conclude the paper.

2. Related Work

Before defining usability attributes for e-learning systems [3], it is necessary to dwell upon the difference between Learning Management Systems (LMSs) and Learning Content Management Systems (LCMSs). An LCMS is a tool to define educational materials that are successively deployed in LMSs. LMSs are integrated learning environments conceived to deploy and enjoy educational materials. Such a scenario highlights that LMSs, LCMSs, and consequently educational materials should be independently analysed, since for each of them different attributes are considered relevant. It is worth noting that the success of e-learning projects often passes through the satisfaction degree that the people perceive on the used LMSs and LCMSs tools [8].

The e-learning literature proposes a good number of tools assisting instructional designer during the analysis, design, implementation, and delivery of instruction via the Web.

Muraida and Spector [10] assert that there is "a lack of instructional designer expertise, pressure for increased productivity of designers, and the need to standardize products and ensure the effectiveness of product". Thus, tools supporting instruction design during all the phases of the learning process definition are desirable. Goodyear [6] views the instruction design as falling within four main approaches. These approaches allow the instruction designer to generate e-learning activities from given specifications by means of tools supporting the design of course structure, the selection of presentation templates, the reuse of design elements, and the coordination of activities accomplished by a design team. Goodyear [7] also proposes an approach for analysing and designing distance courses that is divided into neat parts. The first part of Goodyear's approach resembles the work of other people (outside education) who are interested in the design of technology supporting the work of information systems designers, requirements engineers, human factors specialists, and so on. The second part is instead focused on the design of good learning tasks exploiting traditional analysis and design processes. Often, these tools are not able to compensate the lack of expertise of instruction designers. Vrasidas [11] presents a system to develop hypermedia approaches as part of courses and learning environments delivered on the World Wide Web. It details the structuring of information, branching and interactivity, user interface, and navigation through Web based distance courses. Differently from the approaches above, the AIMS Project [2] describes a Theoretical Framework in which the knowledge domain editing and the course editing are distinguished. First the instructional designer constructs the domain model in term of concepts and links and then the didactic contents are embedded.

Differently from the described tools and methods our proposal aimed at exploiting the ideas and the benefits of component based approaches and visual languages as well. If on one side the visual languages support non expert user during the definition of adaptive learning processes, on the other side the tool supports the instructional designer in the reuse of learning components, at different granularity levels. Moreover, the work discussed so far does not consider the usability in the design of didactic materials.

3. ASCLO-S Language and Tool

In this section we briefly recall the approach proposed in [4] to define and create adaptive learning processes. This approach is based on four different granularity levels of knowledge contents: ASCLO-S, ASCLO, Knowledge Fragment, and Raw Contents. An ASCLO-S provides the higher level, so it can be refined by using a sentence of the ASCLO-S language. The sentences of this language describe aggregations of Adaptive Self Consistent Learning Objects (ASCLO). Each ASCLO is a set of didactic contents that are presented to the student by considering him/her knowledge. It can be considered as a logical collection of Knowledge Fragments and an assessment test. A Knowledge Fragment is composed of Raw Contents, which are multimedia objects presented in linear way to the student.

The visual tokens of the ASCLO-S language are shown in Figure 1. Generally, sentences of this language are composed of a set of Knowledge Fragments (Figure 1A), which are arranged using Joint/Disjoint symbols (Figure 1D) and labelled arrows (Figure 1B). Adaptive learning processes are defined for student classes, which are represented by stereotypes. Both the learners and the stereotypes are implemented with Linguistic Terms based on Linguistic Variables [12].

The instructional designer first specifies learning processes using labelled arrows, and then associates stereotypes to these labels. The learning process that will be presented to the student is the one with the stereotype more similar to his/her profile.

On the other hand, Joint/Disjoint symbol is used for jointing and disjointing the learning process defined for two or more classes of stereotypes. Figure 1C shows the symbol representing the final test that can be presented at the end of each ASCLO. Using the results of the self-assessment tests the student knowledge profile is updated.

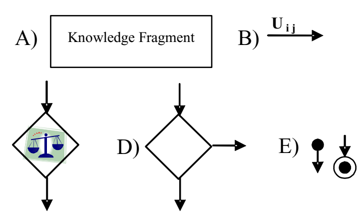


Figure 1. The visual language icons. A) Knowledge Fragment; B) Transition Element; C) Modelling Student Test; D) Joint/Disjoint Element; E)Start/Stop Marker

The language allows the instructional designer to highlight the ASCLOs using dotted rectangles. Moreover, the instructional designers can arrange ASCLOs that have strictly cohesive contents into swimlanes. Finally, the start and the stop markers

(Figure 1E) are used to indicate the initial and final states of the ASCLO-S language.

An example of adaptive learning processes for the XML topic is shown Figure 2. This process has been defined using the ASCLO-S visual environment of the SEAMAN tool [5].

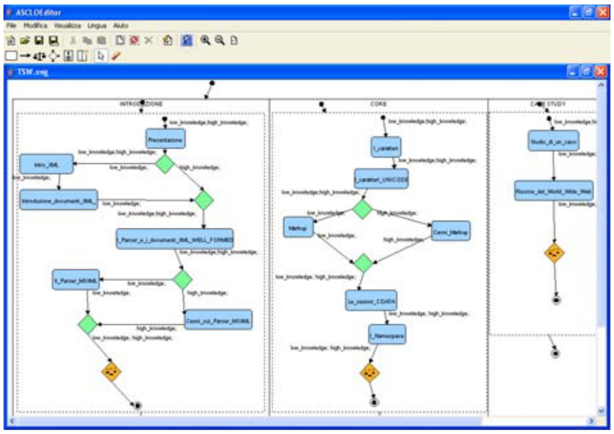


Figure 2. A sentence example in the ASCLO-S editor

4. Usability Test

A pilot test was conducted before the proposed usability test. The pilot test was to try out those equipment and facilities, test the design of experiment sessions including materials used, and methods of collecting data. Expert users were chosen to this pilot test. In general, the contribution of these users allowed improving the ASCLO-S usability before the actual usability test.

For the usability test a group of seven teachers was recruited. All the teachers underwent an introductory course of 1 hour on the ASCLO-S editor and its visual notation. After that they were asked to use the system on their course, not having the possibility to invoke individual tutor support. When the course content design finished the teachers filled in a questionnaire. The usability test was performed from Monday 23rd August 2004 to Thursday 9th September 2004.

The group of teachers used to carry out the usability test was very heterogeneous. In particular, about half of the selected testers were secondary school teachers with different backgrounds, while the others were people with different teaching experiences. Table 1 summarises the profiles of the teachers used in the usability test. The first and the second columns contain the tester and his/her background, respectively. Finally, the age is contained in the third column, while the fourth column contains the sex. The teachers are grouped in four age intervals. It is worth noting that they realised courses or part of them on their subjects.

Once the pilot test was proved to be feasible and effective, the usability test was carried out subsequently. This usability test was performed in one-to-one session using the think aloud rule. All the teachers underwent an introductory course of one hour on the ASCLO-S editor and its visual notation. After that they were asked to use the tool for twenty minutes without invoking the tutor support. Successively, the testers were asked to use the ASCLO-S editor having the possibility to invoke individual tutor support. In particular, the teachers had to entirely design the e-learning course or a part of it performing the following tasks:

- Define the students' stereotypes.
- Insert at least one image.
- Insert at least a hyper textual link and an e-mail address.
- Check the syntax of the sentence you drew.
- Save and export the created course.

Tester	Background	Age	Sex
Teacher1	Student of Computer Science	25 – 34	F
Teacher2	Student of Italian Literature	18 – 24	F
Teacher3	Teacher of Art Education	over 45	M
Teacher4	Manager of a local archaeologist association	over 45	M
Teacher5	Sociologist	25 – 34	F
Teacher6	Teacher of Music Education	35 – 45	M
Teacher7	Teacher of Art Education	over 45	M

Table 1. Recruited Teachers

After having defined the learning process trough the ASCLO-S editor the teachers were asked to fill in a questionnaire aimed at assessing the tool perceived usability. The designed questionnaire was composed of five categories: *Tester Experience*, *General Evaluation*, *Special Judgement*, *Tool Learning*, and *Information Grant*.

The *Tester Experience* category was composed of three questions, which aimed at knowing the background of the statistic sample. The general reaction of the teachers in terms of satisfaction degree was evaluated by the questions belonging to the *General Evaluation* category. The questions of the *Special Judgement* category were used to assess the usability that testers perceived of the tools' graphical user interfaces. The *Tool Learning* category aimed at evaluating the satisfaction degree to master the tools. Finally, the information provided by the tool while the teacher was using it, was evaluated by the questions of the *Information Grant* category. The questions of these categories less the ones belonging to the *Tester Experience* are shown in Table 2.

Id	Question
2.1	The tool is (from *horrible* to *wonderful*)
2.2	Using the tool is (from *difficult* to *simple*)
2.3	The aroused feeling by the tool use is (from *frustrating* to *satisfactory*)
2.4	The tool use is (from *boring* to *exciting*)
3.1	The user interface is (from *irritant* to *pleasant*)
3.2	The tool is (from *complicated* to *simple*)
3.3	The tool proposes error messages (from *vague* to *specific*)
3.4	Creating e-learning courses with this tool is (from *boring* to *exciting*)
4.1	Learning to use the tool is(from *difficult* to *simple*)
4.2	The required time to use the tool is (from *much* to *little*)
4.3	Remembering the command and their use is (from *difficult* to *simple*)
4.4	The number of steps to carry out a task is(from *too much* to *right*)
4.5	The required time to insert a link is (from *much* to *little*)
4.6	The number of steps to insert an image is (from *much* to *little*)
4.7	Defining the stereotype is (from *difficult* to *simple*)
4.8	Exporting the content created using the tool is (from *difficult* to *simple*)
5.1	Icon names and objects have an clear meaning (from *never* to *always*)
5.2	Each set of operations produce a predictable result (from *never* to *always*)

Table 2. Questions of the usability questionnaire

The answer for each question of the usability questionnaire ranges between 1 and 9. For example, the value 1 for the question *2.1* represents the worst judgment (*horrible*) that the tester can express on the tool user interface, while the best judgment (*wonderful*) is expressed by the value 9.

5. Usability Test Results

In this section, the results of the usability test are presented. These results were derived from two sources: the tasks performed by the users and the questionnaires. Thus, mainly the results are organised in two classes of findings: objective and subjective. The objective findings include all of the subjects' response times for the relevant aspects of the usability test, while the results from the questionnaires constitute the subjective findings. The analysis of both objective and subjective findings produced a reports aimed at providing feedbacks to improve the usability of the ASCLO-S editor.

The following are the factors that have been considered relevant for the objective findings: time to explore the tool features, time to carry out the usability test, time spent in inactivity, features never used, tasks not correctly carried out, and the number of mistakes.

The times considered relevant for the objective usability findings are contained in Table 3. In particular, the teachers have spent fifteen minutes as mean time, while ten and twenty minutes are the best and worst value to explore the tool, respectively. The second column presents the time to accomplish the usability test. The time to design the adaptive learning process as well as the underlying learning contents has been considered. A considerable difference between the best and the worst time to accomplish the usability test depend on the time spent to turn the didactic materials only available in hard copy into electronic format. Moreover, such a difference is also caused by the size of the realised didactic contents. For example, although Teacher1 had more teaching material in electronic format than Teacher5 the learning process of the former teacher was more complex, thus the time to accomplish the usability test was greater. The inactivity time is shown in the third column. Teacher1, Teacher2, Teacher4, Teacher6, and Teacher7 worked without stopping until they had finished their tasks, while Teacher3 and Teacher5 were inactive for 1 and 2 minutes, respectively.

Tester	Exploration time	Accomplishment time	Inactivity time
Teacher1	10	240	0
Teacher2	15	59	0
Teacher3	20	45	1
Teacher4	18	65	0
Teacher5	12	35	2
Teacher6	17	58	0
Teacher7	15	63	0

Table 3. Times of the objective usability findings

The features provided by ASCLO-S editor have been all used. On the other hand, Teacher5 and Teacher7 did not master the diagrammatic notation of ASCLO-S language. They never used the Joint/Disjoint element to define their adaptive learning process. It was mainly due to the fact that these two testers were not familiar with flow-diagrams, and so mapping the concept of condition for an activity on real world problems was not immediate. Consequently, the Teacher5 and Teacher7 did not accomplish the task concerned the realisation of an adaptive learning processes. Anyway, all the recruited teachers did not make notable mistakes.

In the case of a typical usability test, however, where the goal is to uncover as many usability problems as possible, descriptive statistics, as mean values, are sufficient. To conclude the usability evaluation of the ASCLO-S editor, the questionnaire answers have been analysed as well. This analysis allows identifying the subjective findings as perceived usability by the recruited teachers. To classify the subjective satisfaction degrees the ranges shown in Table 4 have been used, while the perceived usability of the testers is shown in Table 5. Since nobody of the selected teachers perceived a scant satisfaction degree, then for space reasons the scant column is not contained in Table 5.

In general, the usability questionnaire revealed that the satisfaction degree of the seven teachers is more than sufficient. The *General Evaluation* of ASCLO-S editor was fairly good for 72% of the testers, while

14% of the users expressed a good judgement. Finally, the others users manifested a sufficient judgement.

Range	Evaluation
[1, 2.5[	Scant
[2.5, 4.5[	Insufficient
[4.5, 5.5[	Mediocre
[5.5, 6.5[	Sufficient
[6.5, 7.5[	Fairly good
[7.5, 8.5[	Good
[8.5, 9[	Excellent

Table 4. Ranges to evaluate the perceived usability

Questions	Insuff.	Mediocre	Suff.	Fairly Good	Good	Exc.
2.1	0%	0%	28%	58%	14%	0%
2.2	0%	0%	14%	72%	14%	0%
2.3	0%	0%	28%	72%	0%	0%
2.4	0%	0%	14%	86%	0%	0%
General Evaluation	0%	0%	14%	72%	14%	0%
3.1	0%	28%	44%	28%	0%	0%
3.2	0%	0%	58%	28%	14%	0%
3.3	0%	0%	28%	44%	28%	0%
3.4	0%	0%	14%	86%	0%	0%
Special Judgment	0%	0%	43%	57%	0%	0%
4.1	0%	0%	29%	57%	14%	0%
4.2	0%	0%	43%	43%	14%	0%
4.3	0%	0%	29%	57%	14%	0%
4.4	0%	0%	28%	44%	28%	0%
4.5	0%	0%	0%	57%	14%	29%
4.6	0%	0%	14%	43%	14%	29%
4.7	14%	14%	44%	14%	14%	0%
4.8	0%	0%	0%	29%	42%	29%
Tool Learning	0%	0%	0%	86%	14%	0%
5.1	0%	0%	0%	72%	28%	0%
5.2	0%	0%	14%	72%	14%	0%
Information Grant	0%	0%	0%	57%	43%	0%

Table 5. Perceived usability of the ASCLO-S editor

The *Special Judgment* category aimed at evaluating the users' reactions on the ASCLO-S tool usage. In particular, the questions of this category allow having feedbacks on the satisfaction degree of users' interaction with the tool. Relevant suggestions of the recruited testers have been also gathered. In particular, Teacher3 and Teacher5 found the user interface very lean. They suggested enhancing the graphical user interface with colours to make it more appealing and consequently improve the perceived usability degree.

Better results were achieved in terms of *Tool Learning*. 86% of the users considered the tool fairly good to learn, while a good judgment was expressed by 14% of the users.

Information Grant category revealed the ASCLO-S editor provided an encouraging support while the teacher is using it in general. In particular, 43% of the teachers expressed a good judgment, while a fairly good judgment was expressed by 57% of them. This appreciable result was due to the fact that all the components of the graphical user interface are understandable and the result of each operation produces always a predictable result.

6. Discussion

The teachers appreciated very much the facilities of the ASCLO-S editor to insert multimedia objects and to export the created knowledge contents. Great satisfaction was also manifested when the adaptive learning process was generated and deployed to be delivered in the E-World framework, an LMS developed at University of Salerno. This tool integrates a suitable software component to manage the adaptive learning process flow as well as a Run-Time Environment to trace the student learning processes as suggested in the SCORM reference model [1] by the Advanced Distributed Learning (ADL) Consortium [1]. The recruited teachers declared to have appreciated the results of their work and how the knowledge objects were shown in the Web browser. Encouraging results in terms of time spent to learn and use the ASCLO-S editor have been also obtained. Finally, although the teachers did not manifested any particular problems to learn the tool a significant issue was raised from Teacher3, Teacher5, Teacher6, and Teacher7. These defined the learning process flows without manifesting evident problems, even if the binding between symbolic names and students' stereotypes raised a significant issue. Thus, a simpler method to define adaptive processes was suggested.

The same kind of test has been carried out on the Authorware 7.0 [9] a widespread LCMS of the Macromedia. This usability test was accomplished using the same seven teachers of the ASCLO-S editor usability study and was performed from Monday 13th September to Tuesday 21st of September 2004.

Although the satisfaction degree reached by all the recruited users was good on the average, the visual notation of the ASCLO-S editor was preferred with respect to the one of the Authorware tool. They found the visual environment of the ASCLO-S editor simpler and more appealing. Moreover, even if the teachers preferred the result of the didactic contents created by the ASCLO-S editor the management of multimedia objects was considered very simple for both the analysed tools. The teachers manifested great satisfaction when their didactic contents were delivered in the E-World framework.

A point in disfavour for the Authorware tool is the required efforts to master it. Indeed, the teachers on one side found the basic features provided by the

Authorware tool very simple to learn and understand, while on the other side low satisfaction degree is aroused by its advanced features.

Last but not least, the comparison between the ASCLO-S editor and the Authoware tool revealed that the former provides a better support for not expert users, while the latter is more suited for expert users. In general a specific background and computer science knowledge is required to use the Authorware tool. The usability studies also revealed that non expert user found the ASCLO-S editor simpler to use. Thus, the ASCLO-S editor could be useful in teaching contexts where teachers or lectures do not have specific computer science background.

It is worth noting that further details on the usability test of the Authorware tool have not been provided only for space reason.

7. Conclusions and Future Work

In this paper we have presented the results of a usability study on the ASCLO-S editor. A group of seven teachers with different know-how and teaching experiences has been recruited. The selected teacher created the learning processes without manifesting any kind of listlessness and indifference. It was due to high satisfaction degree aroused from the use of the tool as well as to the value of the visual languages. We have also presented results achieved from the comparison between ASCLO-S editor and Authorware 7.0.

Future work will be devoted to use the ASCLO-S editor on several educative contexts. In particular, our aim will be to assess the adaptive learning processes in blended and pure teaching contexts, thus to analyse the data gathered by the traceability learning process through suitable empirical studies. The empirical studies should provide feedbacks to estimate both the efforts to create and to enjoy adaptive learning processes by instructional designers and students, respectively. Finally, to support the instructional designer in the development of learning process a methodology will be suggested.

References

[1] ADL (2004). Advanced Distributed Learning (ADL) Initiative. *From http://www.adlnet.org.*

[2] AIMS: Adaptive Information System for Management of Learning Content (2004). *From wwwis.win.tue.nl:8080/AIMS.*

[3] C. Ardito., M. De Marsico., R. Lanzilotti., S. Levialdi., T.Roselli., V. Rossano., and M. Tersigli. "Usability of E-Learning Tools". *In Proceedings of the working conference on Advanced Visual Interfaces*. Gallipoli, Italy, 2004, pp.: 80-84.

[4] G. Casella, G. Costagliola, F. Ferrucci, G. Polese, G. Scanniello. "Visual Languages for Defining Adaptive and Collaborative e-Learning Activities". *In Proceedings of IADIS International Conference: e-Society 2004*, Avila, Spain, 16-19 July, 2004, Vol. 1, pp.: 243-250.

[5] G. Costagliola, F. Ferrucci, G. Polese, and G. Scanniello. "A Visual Language Based System for Designing and Presenting E-learning Courses". *In International Journal of Distance Education Technologies*. Vol. 3, No.1, January-March, 2005, pp.: 1-19.

[6] P. Goodyear. "Instructional design environments: Methods and tools for the design of complex instructional systems". *In S. Dijkstra, N. Seel, F.Schott, and R.D. Tennyson, (eds) Instructional Design: International Perspectives*. Vol. 2, Mahwah, NJ: Lawrence Erlbaum Associates, 1997.

[7] P. Goodyear. "Seeing Learning as Work: Implications for Understanding and Improving Analysis and Design". *In Journal of Courseware Engineering*. Vol. 2. 3-11, 1999.

[8] V. King, "A Product-based Model for Successful Online Learning Projects". *In Proceeding of 1st International Conference on Information Technology: Research and Education*, Newark, New Jersey, USA, 2003, pp.: 215-218.

[9] Macromedia, Authorware 7.0, 2005, *from http://www.macromedia.com/software/authorware/*

[10] D. J. Muraida and J. M. Spector. "Automatic Design Instruction". *In S. Dijkstra. N. Seel. F. Schott and D. Tennyson. Instructional Design: International Perspectives*. Mahwah, NJ: Lawrence Erilbaum, Vol. 2, 1997.

[11] C. Vrasidas, "A Systematic Approach for Designing Hypermedia Environments for Teaching and Learning". *In International Journal of Instructional Media,* http://www.cait.org/vrasidas/hypermedia.pdf, 2002.

[12] L. A. Zadeh. "The concept of a linguistic variable and its application to approximate reasoning", *Part III: Inf. Sci.* 9, 1975, pp.: 43-80.

Towards a Multimedia Ontology System: an Approach Using TAO_XML *

Massimiliano Albanese, Paolo Maresca, Antonio Picariello and Antonio Maria Rinaldi
Dipartimento di Informatica e Sistemistica
Università di Napoli "Federico II"
Napoli, Italy
E-mail: {malbanes,paomares,picus,amrinald}@unina.it

Abstract

Archiving, organizing, and searching multimedia data in an appropriate fashion is a task of increasing importance. The ontology theory may be appropriately extended in order to face with this challenging issue. In this paper we propose a novel multimedia ontology theory. We first describe the multimedia ontology concepts and then we adopt TAO_XML *as a suitable ontology description language. Eventually, we propose a general architecture for supporting creation and management of multimedia objects.*

1. Introduction

The rapid evolution of digital technology is producing a tremendous amounts of digital data such as images, music, movies, and other types of media. Thus, the management of multimedia objects is a task of increasing importance for users who need to archive, organize, and search their multimedia collections in an appropriate fashion.

At the present, users typically arrange their multimedia collections into file systems which provide poor naming mechanisms and hierarchical directory structures for organization and searching. In particular, this approach has the following drawbacks:

- the categorization depends on the used classification hierarchies;
- the logical organization strictly depends on the physical storage system;
- identification based on file names alone is often not globally consistent (e.g., duplicates are possible);
- the semantic content of multimedia objects is difficult to represent and manage.

Semantics of digital multimedia materials are very hard to capture either by manual or automatic way: these semantics may be viewed as the set of terms created or linked in the practice, which forms the multimedia ontology of the discourse. Till now, knowledge management has been accomplished by the use of *ontologies*, having their primary area of application in the field of knowledge engineering. An ontology is thus a *terminological abstraction of the real world* [3]. By the way, it's the author opinion that an ontology based representation of multimedia information may greatly improve multimedia systems, in order to structure content and support retrieval.

In this paper we propose a novel multimedia ontology theory. We first describe the multimedia ontology concepts and then we use TAO_XML as a suitable ontology description language. We also propose a general architecture for supporting creation and management of multimedia objects.

The remainder of the paper is organized as follows. Section 2 provides a motivating example that will be used throughout the entire paper, while section 3 discusses related works. Multimedia ontology concepts are introduced in section 4. Section 5 provides a background of the TAO paradigm and language, while section 6 describes the proposed multimedia ontology system. Eventually conclusions are reported in section 7.

2. Motivating Example

Providing content-based information is an important activity for a number of applications. In the world wide web domain, for example, search engines may be greatly enhanced if they can use a conceptualization of the managed data. By the way, we notice that no information about the multimedia content is considered at all, simply because the ontology definition does not allow any kind of *multimedia content information*.

*This work has been carried out partially under the financial support of the Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR) in the framework of the FIRB Project "Middleware for advanced services over large-scale, wired-wireless distributed systems (WEB-MINDS)"

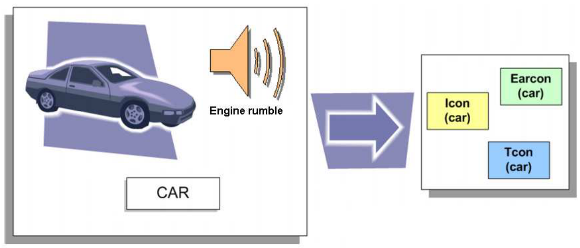


Figure 1. Motivating example

Let us consider, for example, figure 1. From a psychological point of view, humans associate the concept of *car* to either the picture of a car, or the word *car*, or the engine rumble.

Anyway, how to model and represent multimedia data in an ontology system is not a trivial task, especially for the lack of a model that can describe this kind of ontology. Our vision is that of providing a general framework which takes into account not only the textual information, but also the multimedia content of documents.

Let us consider a user who wants to retrieve all the information related to the famous modern painter *Pablo Picasso*. When she submits the query to the search engine, the use of an ontology may surely enhance the retrieval process, thus collecting all the information related to the painter, as the artistic field, the information about the subjects represented in his painting and so on.

In the following of the paper, we will define a model for multimedia ontologies and show a specific implementation using the TAO_XML environment. Figure 2 shows an example of using the TAO_XML representation for the concepts related to *Picasso* (see figure 4 for details).

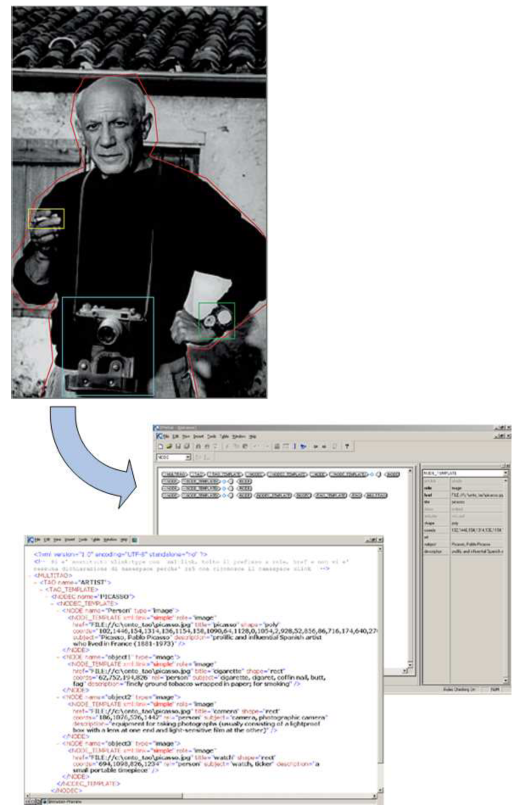


Figure 2. A portrait of Picasso and its representation using TAO_XML

3. Related Works

Multimedia data description and presentation is a hot topic in the research community. In the last years several systems have been presented to provide formal models and languages to address the issues related to the complex nature of multimedia objects. In [2] the authors present Flavor, a formal language for audio visual object representation. The system uses an innovative description, called Flavor, to generate C++ and Java code for describing, processing and producing bit streams according to a specific syntax. The system also provides a framework extension, called XFlavor, in order to offer XML features for media representation.

AROM [13] is an object based knowledge representation system which, together with V-STORM [7] supplies a general framework to manage and describe multimedia data. This paper presents an AROM knowledge base called AVS [5] which represents a generic model for multimedia presentation, using the SMIL standard [14]. The AVS model is described through a graphical notation using UML. Authors of [12] propose a novel framework for describing and indexing multimedia data using a statistical approach. A4SM [11] is a framework for automatic and semi-automatic annotation of multimedia data. The authors use an RDF schema for describing relation based semantics. Different description frameworks as Dublin Core, XML, RDF and so on, are analyzed in [4], where the authors propose a Multimedia Description Framework (MDF), designed to give a unified view of multiple description schemas.

At the best of our knowledge there is no formal definition of what a multimedia ontology should be. In this paper we propose a formalization of a multimedia ontology and propose a language for describing its structure and content.

4. Multimedia Ontology Concepts

It is well known that the word "ontology" generates a lot of controversy in discussions about AI, although it has a long history in philosophy, in which it refers to the subject of existence. It is also often confused with epistemology, which is about knowledge and knowing. We adopt the Gruber idea [3], who argues that the term ontology means a specification of a conceptualization: that is, an ontology is a description of the concepts and relationships that can exist, like a formal specification of a program, providing a shared and common understanding of a domain that can be communicated between people and computer systems.

In this paper we make a first attempt towards the definition of a Multimedia Ontology. The consideration that are at the basis of our idea is that each multimedia object evokes one or more concepts, as any word of a vocabulary does.

In this sense, we can informally define a Multimedia Ontology as a mean for specifying the knowledge of the world through multimedia objects and representing the organization of multimedia documents in a structured way such that users and applications can process the descriptions with reference to a common understanding.

Example 1 *Consider the picture in figure 2: it recalls the concept of 'Pablo Picasso', the famous Spanish painter who depicted a lot of wonderful paintings among which the famous 'Guernica'. The same concept is evoked by the word 'Picasso'.*

We can draw the following considerations from the above example: the first step towards the definition of a Multimedia Ontology requires to extend the concepts of 'word' and 'dictionary'. Starting from this consideration let us now introduce some preliminary and fundamental definitions.

Definition 1 (MM-Alphabet) *A* MultiMedia Alphabet *is a finite set of* MM-Symbols*, where each MM-Symbol is an alphanumerical character, a pixel or an audio sample.*

$$\text{MM-Alphabet} = \{\text{MM-Symbol}\} \tag{1}$$

Two MM-Symbols are said to be homogeneous if they are of the same type.

Definition 2 (MM-Word) *Given an alphabet $\mathcal{A}$, a* MultiMedia Word *of length k over $\mathcal{A}$ is a composition of k homogeneous MM-Symbols from $\mathcal{A}$.*

$$\text{MM-Word}_{\mathcal{A}} = \langle s_1, ..., s_k \rangle,\ s_i \in \mathcal{A}\ \forall\, i \in [1, k] \tag{2}$$

A MultiMedia Word *is said to be* composite *if it can be decomposed into meaningful MultiMedia Words,* atomic *if it cannot be further decomposed.*

Definition 3 (MM-Dictionary) *Given an alphabet $\mathcal{A}$, a* MultiMedia Dictionary *over $\mathcal{A}$ is a set of MM-Words over the alphabet $\mathcal{A}$.*

$$\text{MM-Dictionary}_{\mathcal{A}} = \{\text{MM-Word}_{\mathcal{A}}\} \tag{3}$$

It's worth noticing that the concept of decomposition of MM-Words is different for different kinds of media. In the case of images a component MM-Word is a subregion of the whole picture; in the case of videos a component MM-Word is a subsequence of frames; in the case of text a component MM-Word is a word.

Example 2 *The picture in figure 2 is a composite MM-Word. Clearly the main subject is Picasso, but we can recognize several subregions: the camera, the cigarette and the watches. Each component MM-Word recalls a different concept and contributes to determine the overall meaning of the composite MM-Word.*

Definition 4 (MM-Document) *A* MultiMedia Document *is a composition of MM-Words through a set $\mathcal{R}$ of relations that represents the logical structure of the documents.*

$$\text{MM-Document}_{\mathcal{A}} = (\{\text{MM-Word}_{\mathcal{A}}\}, \mathcal{R}) \tag{4}$$

As a particular case, we notice that, if w is a MM-Word, then $(\{w\}, \emptyset)$ is still a MM-Document.

Definition 5 (Extended MM-Dictionary) *Given a MultiMedia Dictionary $\mathcal{W}$, an* Extended MultiMedia Dictionary *$\mathcal{D}$ over $\mathcal{W}$ is a set of MM-Documents composed of MM-Words in $\mathcal{W}$.*

In the following we will be using the terms *multimedia object* to refer to both MM-Words and MM-Documents.

Definition 6 (Concept) *We define a* Concept *C as a pair $(\mathbb{D}, \mathbb{R})$, where $\mathbb{D}$ is a domain and $\mathbb{R}$ a set of relations between the elements in $\mathbb{D}$. Let $\mathcal{C}$ denote the set of all concepts.*

The elements in $\mathbb{D}$ can be thought as elementary concepts that allow to define a more complex concept. Figure 3 shows an example of such a concept, derived from ConceptNet [6], a freely available commonsense knowledge base and natural-language-processing toolkit which supports many practical textual-reasoning tasks over real-world documents. The concept of *car* is defined through a few simpler concepts connected by several kinds of relations.

We can now define how to map object from the Extended Multimedia Dictionary into concepts.

Definition 7 (Mapping Function) *We define a* Mapping Function *ρ as a function that relates a multimedia object to a specific concept.*

$$\rho : d \in \mathcal{D} \rightarrow C \in \mathcal{C} \tag{5}$$

We say that a mapping function is complete iff

$$\forall d \in \mathcal{D}\ \rho(d) \neq null \tag{6}$$

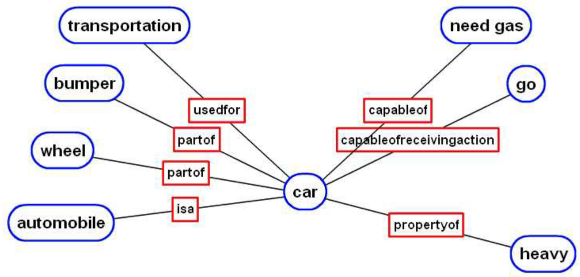


Figure 3. The concept of car from ConceptNet

We say that a mapping function is partial iff

$$\exists d \in \mathcal{D} \mid \rho(d) = null \tag{7}$$

Different mapping functions can assign different meanings to the same multimedia object. In fact each multimedia object, depending on the context, may represent different concepts. The inverse function of ρ returns all the objects that represent a given concept.

$$\rho^{-1} : C \in \mathcal{C} \rightarrow \{d_i\} \in \mathcal{D}^n \tag{8}$$

Given the definition of mapping function we can define the synonymy of multimedia objects.

Definition 8 (Synonymy) *Given a mapping function ρ and two MM-Documents $d_1, d_2 \in \mathcal{D}$, d_1 and d_2 are synonyms w.r.t. ρ iff*

$$\rho(d_1) = \rho(d_2) \tag{9}$$

We remark that the synonymy property holds between terms from the dictionary. In other words two object are synonyms w.r.t. a specific mapping function if they represent the same concept in the domain of that function.

Example 3 *Figure 3 shows an example of different MM-Words – a picture, a sound and a written word – that express the concept of car.*

We can finally give a formal definition of Multimedia Ontology.

Definition 9 (Multimedia Ontology) *We define a* Multimedia Ontology *MO as*

$$MO = (\mathcal{D}, \mathcal{C}, \mathcal{F}) \tag{10}$$

where $\mathcal{D}$ is an extend multimedia dictionary, $\mathcal{C}$ is a set of concepts and $\mathcal{F}$ a family of mapping functions.

Definition 10 (Domain Multimedia Ontology) *We define a* Domain Multimedia Ontology *MO_{dom} as*

$$MO_{dom} = (\mathcal{D}_{dom}, \mathcal{C}_{dom}, \rho_{dom}) \tag{11}$$

where ρ_{dom} is a partial mapping function, $\mathcal{C}_{dom} \subset \mathcal{C}$ is the codomain of ρ_{dom} and $\mathcal{D}_{dom}$ is the subset of all objects $d \in \mathcal{D}$ such that $\rho_{dom}(d)$ is not null.

Example 4 *Figure 2 shows a picture and its* TAO_XML *description. In this picture we notice the presence of several objects which are well represented through our extension of* TAO_XML . *Some elements have been added to the language in order to i) manage spatial and temporal information related, as an example, to subregion of an image or time intervals in an audio file; ii) semantic relations between objects.*

5. TeleAction Objects for Multimedia Ontologies

The second major contribution of this paper is the definition of a model for describing the structure and the content of multimedia ontologies using a suitable language. In this section we introduce the TAO_XML language and show how it can fit the multimedia ontology definitions. In particular we show how the ontology concepts defined so far can be mapped into the language elements.

TAO (TeleAction Object) [1] is a paradigm for representing multimedia objects based on the following two elements: a hypergraph that specifies the component objects and their structural relations, and a knowledge structure which describes the environment and the actions of the object. In this section, we will introduce TAO and show how it can be described using XML, thus opening the way towards the the representation of Multimedia Ontologies.

5.1. Theoretical Background: TeleAction Objects

TeleAction Objects (TAOs) are multimedia objects with an associated hypergraph representing both the multimedia object and the knowledge structure. The knowledge structure allows the TAO to automatically react to certain events. A TAO can be divided into two parts: a hypergraph $\mathcal{G}(\mathcal{N}, \mathcal{L})$ and a knowledge $\mathcal{K}$, where $\mathcal{N}$ is a set of nodes and $\mathcal{L}$ is a set of links. There are two types of nodes: base and composite nodes. Each node represents a TAO and each link represents a relation among TAOs. There are the following link types: (i) attachment, (ii) annotation, (iii) reference, (iv) location and (v) synchronization. Base and composite nodes are called *bundled* when they are grouped, thus defining a single entity. With respect to the formal definitions introduced in section 4, base and composite TAO nodes respectively correspond to atomic and composite MM-Words, while a TAO corresponds to a MM-Document. A whole multimedia system is defined by the MULTITAO element, the root element of the TAO_XML document. A MULTITAO consists of one or more TAOs.

The knowledge structure $\mathcal{K}$ of a TAO is organized into four levels: System Knowledge, Environment Knowledge, Template Knowledge, and Private Knowledge. The knowledge is structured as an active index (IX), which is a set of index cells (IC) from an index cell base (ICB). The index cells define the reactions of the TAO to events filtered by the system. An index cell accepts input messages, performs some action, and sends output messages to a group of ICs. The messages sent will depend on the state of the IC and on the input messages. An IC may be seen as a kind of finite-state machine.

5.2. TAO_XML

The need to represent TAOs through an XML-based language has led to the introduction of TAO_XML [8, 9]. In general, multimedia systems may be viewed as consisting of a set of connected and interacting elementary TAOs. Each TAO is obtained by constructing an hypergraph whose nodes are attached to the index cells which provide the knowledge necessary to the system to react to external events. The hypergraph contains base and composite nodes which are connected via links that describe the relations between the components nodes of the TAO. TAO_XML links are classified as structural, temporal and spatial and correspond to the location, synchronization and annotation and reference TAO links. The attachment link is not used in TAO_XML since the attachment relation is implicitly described by the structure of the document.

5.3. TAO ontologies

We have already seen as the concepts introduced in section 4 can be strictly mapped into the elements of the TAO paradigm. The TAO_XML is thus highly suitable for describing multimedia ontologies. In particular we have extended the language to address some specific issues, adding elements useful to manage spatial and temporal information related, as an example, to subregions of an image or time intervals in an audio file. We have also introduced elements useful to manage semantics of the objects and semantic relations among them. To this aim we use WordNet [10] as a uniform way for representing concepts. Figure 4 shows an example of TAO_XML document, describing the picture of Picasso.

6. Building a Multimedia Ontology System: the architecture and the process

In this section we describe the architecture of the system that has been prototyped at the University of Napoli *Federico II* and currently in the experimentation phase.

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE MULTITAO (View Source for full doctype...)>
<MULTITAO>
 <TAO name="ARTIST">
  <TAO_TEMPLATE>
   <NODEC name="PICASSO">
    <NODEC_TEMPLATE>
     <NODE name="Person" type="image">
      <NODE_TEMPLATE title="Picasso" role="image" href="FILE://c:/tao/picasso.jpg"../>
     </NODE>
     <NODE name="object1" type="image">
      <NODE_TEMPLATE title="cigarette" role="image" href="FILE://c:/tao/picasso.jpg"../>
     </NODE>
     <NODE name="object2" type="image">
      <NODE_TEMPLATE title="camera" role="image" href="FILE://c:/tao/picasso.jpg"../>
     </NODE>
     <NODE name="object3" type="image">
      <NODE_TEMPLATE title="watch" role="image" href="FILE://c:/tao/picasso.jpg"../>
     </NODE>
    </NODEC_TEMPLATE>
   </NODEC>
  </TAO_TEMPLATE>
 </TAO>
</MULTITAO>
```

Figure 4. An example of TAO_XML document

Figure 5 shows the main processes provided by the system, and described in the following:

- *Reverse Document Production - RDP*: it is composed by an initial extraction phase - which takes care of the identification of the basic TAO components - and a second abstraction phase, in which we adopt a representation that is independent from any particular application. The resulting document contains all the basic components and their interconnections. A possible abstraction is a labelled tree as we can see in the example in figure 4. In the example, we want to associate a concept to a multimedia object. In order to do that, we perform multimedia dictionary association and concept association. Once both the associations have been performed, the document is stored into a repository and opportunely indexed using a relational $DBMS$ (Oracle 8i, in our case).

- *Direct Document Production - DDP*: the document extracted from the repository is reconstructed in all of its parts and transformed into its original form. In order to do this, it is necessary to have a subprocess analogous to the one described before.

The architecture has been realized by reusing some tools such as $XMetal$, or implementing new tools at our laboratories, such as transformation, abstraction and extraction tools.

An example of the TAO_XML result of the RDP phase is shown in figure 2. Actually, that is a complete document, which define the component objects and their role in the whole structure.

7. Discussion and Conclusions

The approach we have presented in this paper demonstrates the following advantages:

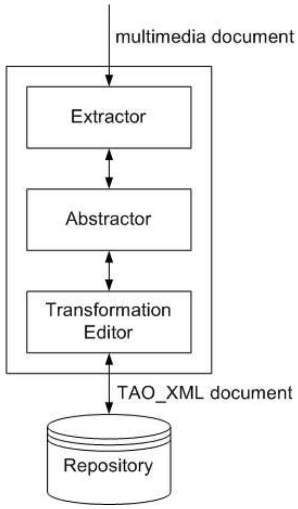


Figure 5. System architecture

- the first obvious advantage is the portability on different hardware/software platforms and the extreme interoperability; the end user, in order to interact with the document collection, only needs a simple browser (e.g. Internet Explorer, etc.);
- the second advantage is the fact that XML is already integrated with the most popular database management systems (e.g. Oracle 8i);
- third, XML is a widespread language in the Internet, thus the documents represented in this format are no longer limited to use in a single organization but may be distributed to other organizations as well;
- eventually, the possibility to share several objects coming from different databases and different organizations may help to build distributed ontologies.

References

[1] H. Chang, S. K. Chang, T. Hou, and A. Hsu. The management and applications of Tele-Action Objects. *ACM Journal of Multimedia Systems*, 3(5-6):204–216, 1995.

[2] A. Eleftheriadis and D. Hong. Flavor: a formal language for audio-visual object representation. In *Proceedings of the 12th Annual ACM International Conference on Multimedia*, pages 816–819. ACM Press, 2004.

[3] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, June 1993.

[4] M. J. Hu and Y. Jian. Multimedia description framework (MDF) for content description of audio/video documents. In *DL '99: Proceedings of the fourth ACM conference on Digital libraries*, pages 67–75. ACM Press, 1999.

[5] M. H. Ketfi A., Gensel J. An object-based knowledge representation approach for multimedia presentations. In *MULTIMEDIA '01: Proceedings of the ninth ACM international conference on Multimedia*, 2001.

[6] H. Liu and P. Singh. Commonsense reasoning in and over natural language. In *Proceedings of the 8th International Conference on Knowledge-Based Intelligent Information & Engineering Systems (KES'2004)*, 2004.

[7] R. Lozano, M. E. Adiba, H. Martin, and F. Morcellin. An object dbms for multimedia presentations including video data. In *ECOOP '98: Workshop ion on Object-Oriented Technology*, pages 553–554. Springer-Verlag, 1998.

[8] P. Maresca, T. Arndt, and A. Guercio. Unifying distance learning resources: The metadata approach. *Journal of Computers*, 13(2):60–76, November 2001.

[9] C. Marmo and P. Maresca. TAO_XML. Master's thesis, University of Napoli, 1999.

[10] G. A. Miller. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, November 1995.

[11] F. Nack and W. Putz. Designing annotation before it's needed. In *MULTIMEDIA '01: Proceedings of the ninth ACM international conference on Multimedia*, pages 251–260. ACM Press, 2001.

[12] A. P. Natsev, M. R. Naphade, and J. R. Smith. Semantic representation: search and mining of multimedia content. In *KDD '04: Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 641–646. ACM Press, 2004.

[13] M. Page, J. Gensel, C. Capponi, C. Bruley, P. Genoud, D. Ziebelin, D. Bardou, and V. Dupierris. A new approach in object-based knowledge representation: The AROM system. In *Proceedings of the 14th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, pages 113–118. Springer-Verlag, 2001.

[14] W. Recommendation. Synchronized Multimedia Integration Language (SMIL 2.0) [2nd Edition]. http://www.w3.org/TR/2005/REC-SMIL2-20050107/, January 2005.

The Image Stack Stream Model, Querying, and Architecture

Alfonso F. Cárdenas Raymond K. Pon Bassam S. Islam
Computer Science Department, University of California at Los Angeles
3731 Boelter Hall, UCLA
Los Angeles, California
{cardenas, rpon, bassam}@cs.ucla.edu

Abstract

Rising volumes of multimedia are being gathered with the increasing deployment of sensors. We present the Image Stack stream model/view of data for querying and visualizing the streaming data. This view is independent of the presence of a DBMS, since more sensors will capture data on a real-time basis. We outline requirements for modeling and visualizing streaming multimedia data with motivating queries. We present the Image Stack data model, a high-level query language for the model, and a system architecture and design to support these requirements. We provide highlights of a prototype implementation in Java and Java Data Objects, bypassing the use of a DBMS as permanent storage.

1. Introduction

Growing types and volumes of multimedia data (alphanumeric, image, sound, and video) are being captured by increasing deployment of sensors (environmental, geophysical, medical, etc). We address the challenge of querying and visualizing of information from multiple streams of different but related types of data, focusing on the access and presentation of data through time. Recent work by others has been reported on video indexing and accessing by content and visual languages [1]-[2]. However, it has generally focused on viewing individual video streams and not on the multiple heterogeneous streams that we are addressing. Furthermore, because much of the expected data streams are multidimensional, using existing streaming data management technology to answer spatio-temporal queries over multidimensional real-time and archived data is difficult. An example of such a query is:

> *"Display the locations of intersections of the UCLA boundaries with Westwood Blvd and Sunset Blvd where the poison fume level exceeds value Y now."*

A stream is an ordered sequence of frames or values. A frame could be an image, a photograph, a frame in a video stream, a text report, or an alphanumeric record changing through time. Current DBMS's (relational or object DBMS's) deal well with alphanumeric record type structures once they are stored and loaded into a database. Unfortunately, it is impractical to store in a DBMS the voluminous data that is arriving rapidly from many sensors.

There has been research regarding alphanumeric data

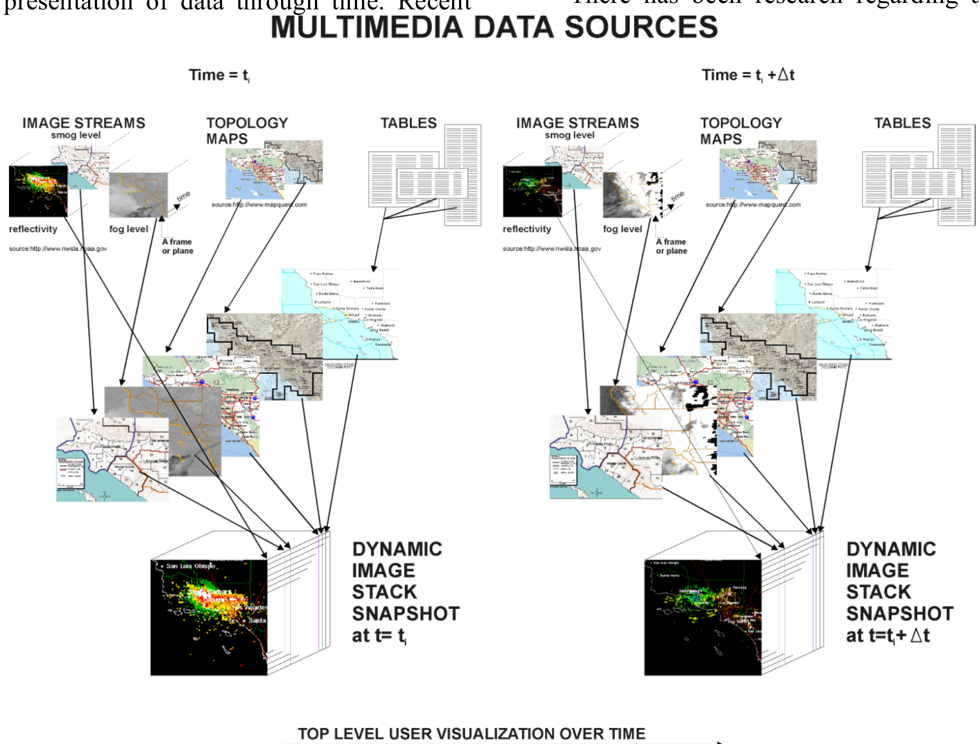


Figure 1: Stream of two Image Stacks.

stream processing. The Cougar project [3] manages data from sensor database systems by using abstract data type functions to represent sensor devices. The NiagraCQ project [4] has focused on retrieving XML data, querying, and monitoring them for some interesting change, the optimization of queries by grouping similar queries together, and on optimization based on the rate of arrival of data items. The STREAM project [5] has developed a prototype of a data stream management system with support for typical relational DBMS's. The TinyDB project [6] has developed continuously adaptive continuous query techniques, in-sensor-network aggregation of data, and visualizations for sensor data. Like the STREAM project, the TelegraphCQ project [7] has also developed a general system to process continuous queries over data streams. The Aurora Project [8] has focused on optimizations for real-time data stream processing.

In contrast to other streaming data management projects and the advances introduced in [9]-[10], we make the following advances with the Image Stack model:

- A high-level query language enabling *visualizing and querying* historical data with real-time data.
- The ability to query *massive multi-dimensional* data that arrive from a data stream.
- Language semantics allowing querying of absolute and relative points in time of a data stream, encapsulating historical and real-time data as a *single continuum*.

Like ours, prior projects have a query language for querying streaming data. NiagaraCQ's language is based on XML; whereas, the other projects are based on SQL. We present a query language, isOQL, based on ODMG OQL [11], intended for use upon data streams. Borrowing notions such as time windows from otherdata stream management projects, isOQL also provides for visualizing and querying historical data.

Other systems have focused on querying simple tuples from continuous streams while the Image Stack focuses on querying massive multidimensional streaming data. We provide an architecture for processing, querying, and visualizing multidimensional multimedia data. Being a data model for high-level querying and visualizing streaming and archived data, the Image Stack has not focused on optimizations for streaming alphanumeric data, but can leverage existing stream processing technologies to take advantage of such optimizations.

Many of the mentioned projects have focused on managing streaming real-time data along with some stored relational data, but have treated historical data as a separate entity. The semantics of CQL [12] and similar stream-based languages are based on querying data from a real-time stream and treat historical data as a different entity (i.e., a separate table). However, we argue that this approach is not intuitive. For example, from a user's perspective, current smog data and smog data from 10 years ago are both instances of smog data, and so the two should not be differentiated. Sliding window semantics of existing stream-based languages may provide some historical data within the same stream being queried in real-time, but data that is available before the query has been issued or falls outside the sliding window of the query is not available for querying. Only the Aurora project has addressed this issue by introducing connection points to store and process historical along with streaming data. In [7], CACQ encapsulates historical and real-time data as a single entity to support disconnected operation, allowing users to register queries and return intermittently to retrieve the latest answers, by applying old data to new queries and then new data to old queries, when new data becomes available. isOQL takes a similar approach, but unlike CACQ, isOQL provides for the explicit specification of an absolute historical point in time in a data stream, in addition to the current time and points relative to the current time.

We have proposed the *Image Stack model/view* as an attractive way to visualize multiple types of data and set up as a local or user view database to support major types of multimedia queries. Figure 1 shows an example of the Image Stack which consists of several planes. Each plane contains a different type of two-dimensionally encoded data that are co-registered to the same coordinate system. This stack is be composed of elements at a point in time from different data streams. The example also shows how several data sources from which data at one point in time could be gathered logically and viewed as planes in the Image Stack. In many multimedia applications, such as environmental analysis, the main interest is visualizing the changes and trends. This leads us to introduce the notion of a *stream of Image Stacks*, also illustrated in Figure 1 for a stream with two stacks showing change through time.

Figure 2 illustrates the basic data model of the Image Stack. Conceptually, a Stack object contains many Frame objects, which are simply represented as a two dimensional grid of Cell objects. For example, each Cell object may contain a smog value at different locations and be collectively grouped together with a Frame object. A Stack object may contain a Frame object that contains smog level data, a Frame object that contains population density data, a Frame object that contains temperature data, and other Frame objects containing other types of data over the same region. Each of these Frame objects are co-registered so a Stack object may be represented as a three-dimensional cube. We later extend this basic data model to support explicit relationships for all levels of the data model (e.g.,

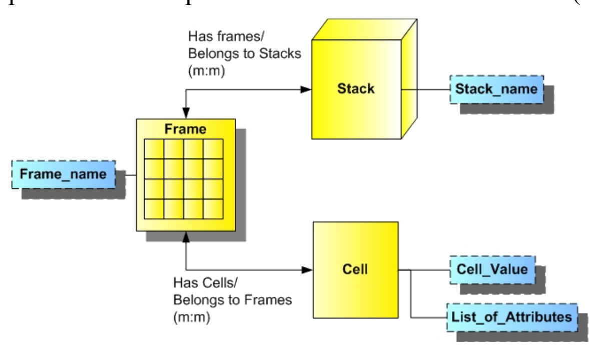


Figure 2: High-level Image Stack model

stack, frame, or cell) and to support temporal querying..

2. High-Level Queries for Multidimension Visualization

We have developed a high-level query language, isOQL, based on ODMG 3.0 OQL for its native object orientation and encompassing of SQL's features, including extensions for operations provided by the Image Stack model.

2.1 Querying Language Grammar

We have used the ODMG OQL as the basis for isOQL, with the adition of several operators and semantic additions to support the Image Stack model. We have created a grammar that is disjoint from the OQL BNF. Our grammar branches off the OQL BNF at the highest level with the following rule:

Query := selectExp | *newQuery* | *expr*

This rule is the same as the OQL BNF, with the addition of the *newQuery* expression:

*newQuery :=displayExp {***OVERLAY** *displayExp}*
*displayExp :=***DISPLAY** *displayList newFromClause [whereClause]*

All of the queries that make use of the visualization features of the query language start with the **DISPLAY** keyword. This allows us to keep queries that use visualization operations separate from traditional OQL SELECT statements. The **OVERLAY** keyword allows two or more visualization queries to be superimposed on one another in a single display screen.

displayList :=displayAttribute {, displayAttribute}
displayAttribute := **(** *displayFunction* **)**

After the DISPLAY statement, the display function follows enclosed in parenthesis. We currently have three classes of display functions: plot functions, cine functions, and contour functions.

displayFunction := plotFunctions | *cineFunctions* | *contourFunction*

The plot functions, defined below, simply plot the cells of the image in a user-selectable color.

plotFunctions := **PLOT_VALUE** *expr [***IN** color*]*
| **PLOT_POINT** *plotLocation [***IN** color*]*
| **PLOT_CHANGE** *expr, expr chageColorExpr*

If the user does not select a color, the system will select a color from a pool of default colors. The **PLOT_VALUE** function plots the value of the current cell value in a gradient color that corresponds to the value of the cell. The **PLOT_POINT** function uses the **LOCATION** keyword to plot the current cell that is being examined. The **PLOT_CHANGE** function takes two cell values as inputs (two cells from the same image at two different time periods), and plots the increases in one color gradient, the decreases in another color gradient, and optionally plots no change in yet another color.

cineFunctions := **CINE** *expr {, expr}*
| **CINE_CHANGE** *expr {, expr} changeColorExpr*
| **CINE_PERCENT_CHANGE** *expr {, expr} changeColorExpr*
changeColorExpr := **INCREASES IN** color, **DECREASES IN** color *[,* **NOCHANGE IN** color*]*

All cine functions output a number of images in an animation. These images are generally the same images at different points of time. These animations help visualize the change of a certain parameter over the same area at different points of time. The **CINE** function takes a number of images as input, and simply displays a slideshow of the images. The **CINE_CHANGE** function takes as input a number of images (of the same area), and shows an animation of the images, displaying the increases between images in one color gradient, and decreases in another color gradient. The **CINE_PERCENT_CHANGE** is identical to the CINE_CHANGE function, except that the change between images is normalized to a percentage.

contourFuction := **CONTOUR_VALUE** *expr* **EVERY** *expr* **IN** color

The **COUNTOUR_VALUE** function plots contour lines on the image, at a user-definable numeric interval, with a user-definable color.

newFromClause := **FROM** *newInteratorDef {,newIteratorDef}*
newIteratorDef := iteratorDef | *streamDef*

Our high-level language queries also include extensions in the traditional FROM clause to better handle stream and image data. The new addition is the *streamDef:*

streamed **:=** *expr[* **[***windowExpr***]** *]* **AS** Alias

The *streamDef* rule allows us to select multiple images in the FROM clause, whether they contain different kinds of data, or the same data at different intervals of time. For the latter case, we have several options for selecting the time periods for the desired images:

windowExpr := windowInstance | *windowInterval* | *windowIntervalFreq*

To select a single snapshot of image data at a certain point in time, we would simply use the *windowInstance* rule:

windowInstance := timeExpr

To select all of the images in a time interval, we would use the *windowInterval* rule:

windowInterval := timeExpr **TO** *timeExpr*

If we wish to select images in a time interval at certain periodic intervals (e.g., images between 2000 and 2002, taken every two months), we would use the *windowIntervalFreq* rule:

windowIntervalFreq :=timeExpr **TO** *timeExpr* **EVERY** *unitExpr*

The following rule defines the allowable expressions of time (e.g., January 2002, **NOW**, or **NOW** – 10 years):

timeExpr := **NOW | NOW –** *unitExpr* **| NOW** + *unitExpr* **|** *timeLiteral*

We use the keyword **NOW** to indicate the most current snapshot of the image data. We also allow mathematical expression to specify a time period relative to NOW. This mathematical expression can reference events in the past and can reference events in the future that may be extrapolated from current data through processes such as regression. Furthermore, we allow the specification of a *timeLiteral* (e.g., "January 1, 2001") to indicate an absolute point in time, as opposed to a relative point in time. Further research is being conducted into defining bounds on how stale the NOW data is allowed to be, and how often the NOW data needs to be refreshed relative to the frequency of change of the data for it to remain "fresh."

2.2 Example Queries

We present several motivating queries, which are typical of the type of major decision-making queries that are not automated today by any generalized DBMS, Geographical Information systems, or visualization system. We show the isOQL incarnations for four queries. The results of the queries would be sent to an imaging or visualization system for display, which are beyond the scope of this paper.

Example 1: *Display the locations of intersections of UCLA boundaries with Westwood Blvd and Sunset Blvd where the poison fume level exceeds value Y now.*

DISPLAY (**PLOT_POINT LOCATION**)
FROM Road **AS** R, School_Boundaries **AS** SB,
    Poison_Levels_Plane_Stream[**NOW**] **AS** PL
WHERE SB.Name **LIKE** "UCLA" **AND**
    (R.Name **LIKE** "Westwood Blvd" **OR**
    R.Name **LIKE** "Sunset Blvd") **AND**
    (SB **INTERSECTS LOCATION**) **AND**
    (R **INTERSECTS LOCATION**) **AND**
    PL.Cell_Value > Y

In the FROM clause, we are examining Road and School_Boundaries which are extents containing objects representing information about roads and school boundaries. Poison_Levels_Plane_Stream is a stream of planes, or a plane stream, as defined in the previous section, indicating time-varying poison levels over a geographic area. This is simply an array of Planes, indexed by time and where each instance of a Plane is a geographical map indicating poison levels. The **LOCATION** keyword indicates the current location of the query execution. The **LOCATION** construct is analogous to the current tuple being processed in relational DBMS's. The NOW keyword can be used as an index in a stream, retrieving the most current plane from the stream. In the preprocessing phase of the query, the NOW symbol is replaced by a numerical index of the most recent plane.

Example 2: *For locations with smog levels over X, show elevation with contour lines every 25 feet for places where there are school districts, showing smog level in purple.*

DISPLAY (**PLOT_VALUE** S.Cell_Value **IN**
    PURPLE), (**CONTOUR** E.Cell_Value **EVERY**
    25 **IN BLACK**)
FROM Smog_Plane_Stream[**NOW**] **AS** S,
    School_Districts_Plane **AS** SD,
    Elevation_Plane **AS** E
WHERE SD.Cell_Value <> **NULL AND**
    S.Cell_Value > X

The DISPLAY command merely displays the result of the query on the viewing device. The PLOT_VALUE function paints the current location of the query execution with a shade of the specified color related to the Cell_Value of the input parameter. The CONTOUR function draws contour lines given the color of the line, the value to contour over, and the distance between contour lines.

Example 3: *Compare the current smog level to a year earlier showing clearly the differences in red for higher smog levels and in blue for lower smog levels.*

We can answer the above query by OVERLAYing the results of two queries, utilizing the OVERLAY operation, but it is expected that the above query form would be common and to rewrite such a complex query for different parameters and different streams would be tedious; thus, we use the display function, PLOT_CHANGE as the following isOQL query illustrates:

DISPLAY (**PLOT_CHANGE** S1.Cell_Value,
    S2.Cell_Value **INCREASES IN RED**,
    DECREASES IN BLUE)
FROM Smog_Plane_Stream[**NOW**] **AS** S1,
    Smog_Plane_Stream[**NOW**–1 Year] **AS** S2

Example 4: *Compare smog level to two, four, six, eight and ten years earlier and show a stream of images clearly indicating for each image the differences in red for higher smog levels and in blue for lower smog levels compared to the prior period; provide also the percentage change in population density and ethnic mix versus the prior period.*

DISPLAY (**CINE_CHANGE** S **INCREASES IN**
    RED, DECREASES IN BLUE), (**CINE** E)
    (**CINE_PERCENT_CHANGE** P **INCREASES**
    IN GREEN, DECREASES IN PURPLE),
FROM City_Areas **AS** CA
    Smog_Plane_Stream[**NOW**-10 Year **TO NOW**
    EVERY 2 Year] **AS** S,
    …
WHERE CA.name = 'East Los Angeles' **AND** CA
    CONTAINS LOCATION

We could have explicitly written conditions for displaying increases and decreases; however, we chose instead to simplify the queries by using **CINE_CHANGE**, **CINE_PERCENT_CHANGE**, and **CINE**. In this query,

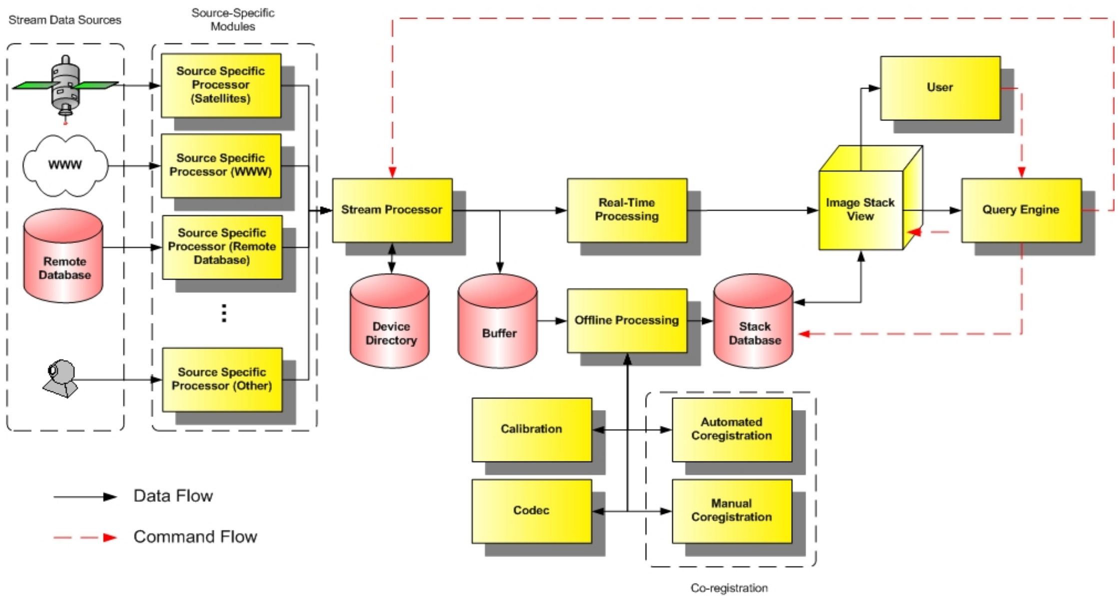


Figure 4: System architecture.

we see that from the base streams, we can select a desired substream for any desired time interval and frequency.

3. System Architecture and Design

3.1 Heterogeneous Data Sources

Figure 4 shows the overall system architecture that will incorporate heterogeneous data sources, real-time and offline processing of data, and the Stack view and querying. The Stream Processor serves as the entry point for stream data into the system. Heterogeneous data sources are handled by special sub-processors that are spawned from the stream processor and are able to handle the extraction of data from each type of source. The drivers for extracting each of the possible input data stream type is stored in a device database. The heterogeneous tuples are then forwarded from the sub-processors to the main Stream Processor, which would then forward the data for further processing. This procedure homogenizes the heterogeneous data into a single framework, so it can be easily manipulated by the rest of the system.

The Stream Processor, the real-time, and offline processing modules are independent modules. This architecture is very much similar to the mediator architecture originally introduced in [24]. This allows for extensibility and flexibility. If a new data stream source is made available, a new source-specific module can be added to make the data source available to users with little modification to the rest of the system. Furthermore, the source-specific modules hide much of the source-specific interfaces from the rest of the system, making system implementation easier and much more modularized. Also, if new processing techniques are developed, they can be added to one of the processing modules as a sub-module.

3.2 Offline versus Online Processing

Note that data streams will not necessarily reside in a conventional DBMS since a majority of sensor data will be too voluminous and will use the Internet as the primary means of providing such data. The Stream Processor may retrieve data continuously real-time, on an ad-hoc basis or on a predefined schedule.

If a DBMS is available that can support the Image Stack model, then its role is shown as the path on the bottom in Figure 4. Data is captured by the Stream Processor and stored temporarily in a buffer. Once the data is stored in the buffer, the data may be compressed, decompressed, calibrated, or co-registered in offline processes via the Offline Processor. The results of the offline processing are stored into the Stack Database so that when the stack view

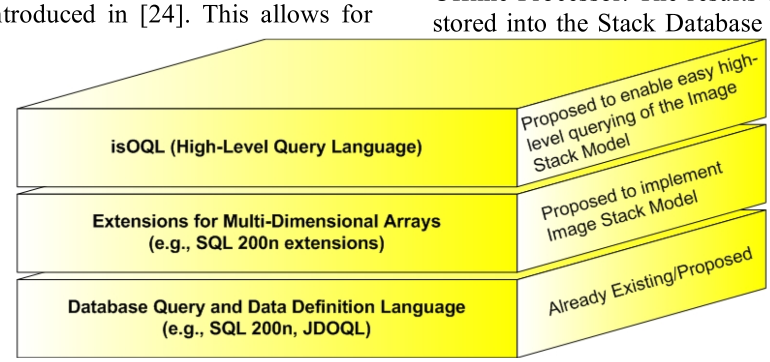


Figure 3: Possible layering of languages.

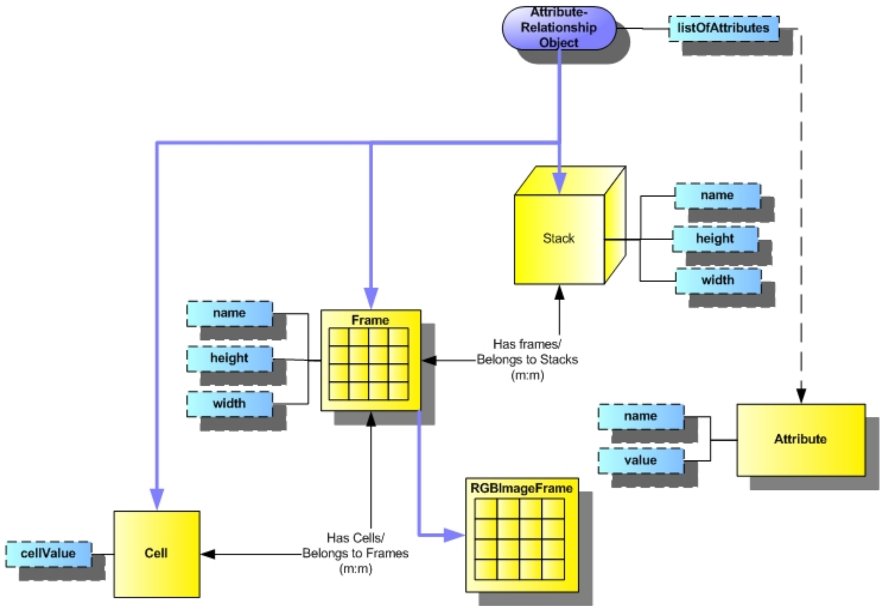


Figure 5: Class architecture for JDO implementation.

is generated later, such processing need not to be done later.

The stream processor, when used in the offline path, will selectively determine which data items should be stored in the DBMS since the data source may provide an infinite stream of data, which cannot all be stored. This selection of data may be based on frequency-based sampling or criteria-based sampling defined by the user or on an on-demand basis, which is an approach taken in the TinyDB project [6]. The selection of data may also be based on identifying which streams have particular characteristics that are of interest, employing methods such as the use of the Hamming Norm [15] or clustering techniques [16]. Other sampling and caching techniques and criteria that may be adapted for the system are discussed in [17] to [20]. An approximation of the data stream or summary data structure may be also employed, such as wavelets [21][22], histograms [23], or sliding windows to reduce the amount of data stored.

If no DBMS is involved, then we can take the streams directly through a real-time stream processing phase via the Real-Time Processor and then provide the Image Stack stream view. Unlike in offline processing, in the real-time processing, the streams are co-registered with possibly less accuracy, depending on the availability of fast algorithms and hardware, which may employ heuristics or approximations and is beyond the scope of our work.

By separating the processing of data into two paths: real-time and offline, data can be processed and viewed on a real-time basis or may be viewed later. Thee data flow path is split into these two paths because we may have different algorithms in processing real-time data and data to be stored in the DBMS in regards to performance, speed, and accuracy.

3.3 Stack View Management and Querying

The Image Stack View is the main interface for the user to view and access the stack data. Queries and data manipulation in the Image Stack View takes place via the Query Engine module. A user issues an isOQL query on the data in the Image Stack View, which would send the query to the Query Engine. Figure 3 shows the layering of the database languages made use by the system. At the top layer is isOQL, which is used to interact with the Image Stack model. Below isOQL are extensions needed to the existing DBMS to handle multi-dimensional data. Beneath the extension layer are the existing database languages, such as JDOQL. The language layering allows isOQL and the Image Stack view to be used in a variety of environments. The high level query language, isOQL, is used so that it would be easier for the user of the system to query it, hiding much of how the Image Stack model is implemented in the DBMS. isOQL incorporates the real-time and DBMS aspects of the system.

The Query Engine determines whether the data necessary to answer the query is present in the Image Stack View, or if more data would need to be loaded from the Stack Database, or if real-time data needs to be requested from the Stream Processor. If necessary, it sends commands (shown in dashed arrows in Figure 4) to the Stack Database to load the data into the Image Stack View and executes the query there. If the query requires real-time data, the Query Engine sends a request for data to the Stream Processor and sends the data to the Image Stack View via the Real-Time Processor. It is possible that the user would wish to use the output of a query as one of the inputs to another. Since the results of all queries are initially stored in the Image Stack View, the user may store the query results in the Stack Database.

4. System and Data Model Implementation

Figure 5 shows the class structure diagram used in Java to implement the above functionality. A Stack, Frame, and a

Cell are subclasses of special super-class, called the Attribute-Relationship Object. An Attribute-Relationship Object has a dynamic list of Attribute objects. Since the Stack, Frame, and Cell classes are Attribute-Relationship Objects, they have a list of Attributes, so any number of user-defined attributes can be added to the whole Stack, Frame or single Pixel, providing storage of data at all levels.

We have implemented the data model in Java Data Objects (JDO) [26], making use of the OO-DBMS FastObjects t7 [27]. We have a preliminary implementation of isOQL built on top of JDOQL, which is demonstrated at [13], using land surface temperature data from the MODIS instrument [14] aboard the Terra satellite, which is already co-registered. Future work will adapt advanced co-registration methods such as [25] if co-registration data is not available.

5. Conclusion and Future Work

We are pursuing further development of the Image Stack view along with streams of stacks over a multitude of multimedia streams. The intent is to provide such a view over data whether or not a DBMS is used at all. We highlight an isOQL engine and Image Stack system without use of a DBMS since many of the data streams broadcasted via the Internet will not reside in a DBMS.

Acknowledgement

This work is partially supported by National Science Foundation Grant # IIS 0140384, "Multimedia Stream Modeling, Relationships and Querying."

References

[1] Flickner M (1995) Query by Image and Video Content: The QBIC System. Computer 28(9)

[2] Chang S, Chen W (1997) VideoQ: An Automated Content Based Video Search System Using Visual Cues. ACM Multimedia

[3] Bonnet P, et al (2001) Towards Sensor Database Systems. 2nd Int'l Conference on Mobile Data Management, Hong Kong.

[4] Chen J, et al (2000) NiagaraCQ: A Scalable Continuous Query System for Internet Databases. SIGMOD 2000: 379-390

[5] Motwani R, et al (2003) Query Processing, Resource Management, and Approximation and in a Data Stream Management System. CIDR 2003, Pacific Grove, California: 245-256

[6] Madden S R, et al (2003) The Design of an Acquisitional Query Processor for Sensor Networks. SIGMOD, San Diego, CA

[7] Chandrasekaran S, et al (2003) TelegraphCQ: Continuous Dataflow Processing for an Uncertain World. CIDR 2003

[8] Carney D, et al (2002) Monitoring Streams - A New Class of Data Management Applications. VLDB Conference, Hong Kong

[9] Cárdenas A F and Michael P A (2002) Image Stack Model of Multimedia Data. DMS 2002, San Francisco, CA.

[10] Cárdenas, A F, et al (2003) Image Stack Viewing and Access. Journal of Visual Language and Computing 14: 421-441

[11] Cattell R G (2000) The Object Database Standard: ODMG Release 3.0, Morgan Kaufmann, San Francisco, CA

[12] Arasu A, Babu S, Widom J (2003) An Abstract Semantics and Concrete Language for Continuous Queries over Streams Relations. International Conference on Data Base Programming Languages

[13] MMSS Group (2003) Geophysical TimeLine [Online] Available: http://www.mmss.cs.ucla.edu/GeoHome.htm

[14] NASA (2003) MODIS Web [Online] Available: http://modis.gsfc.nasa.gov/

[15] Cormode G, et al (2003) Comparing Data Streams Using Hamming Norms (How to Zero In). Transactions on Knowledge and Data Engineering 15(3): 529-540

[16] Guha S, et al (2003) Clustering Data Streams: Theory Practice. Transactions on Knowledge and Data Engineering 15(3): 515-528

[17] Cho J, Garcia-Molina H (2000) Synchronizing a database to Improve Freshness. SIGMOD

[18] Cho J, Ntoulas A (2002) Effective Change Detection using Sampling. VLDB Conference, Hong Kong, China

[19] Olston C, Widom J (2002) Best-Effort Cache Synchronization with Source Cooperation. ACM SIGMOD, Madison, Wisconsin

[20] Olston C, Jiang J, Widom J (2003) Adaptive Filters for Continuous Queries over Distributed Data Streams. SIGMOD, San Diego, California

[21] Gilbert A, et al (2001) Surfing Wavelets on Streams: One-Pass Summaries for Approximate Aggregate Queries. VLDB Conference: 79-88

[22] Gilbert A, et al (2003) One-Pass Wavelet Decompositions of Data Streams. Transactions on Knowledge and Data Engineering 15(3): 541-554

[23] Guha S, et al (2001) Data Streams Histograms. Symp Theory of Computing: 471-475

[24] Wiederhold G (1992) Mediators in the Architecture of Future Information Systems. IEEE Computer: 38-49

[25] Chen Y, et al (2002) Efficient Global Optimization for Image Registration. IEEE Transactions Knowledge and Data Engineering 14(1): 79-92

[26] Sun Microsystems (2003) Java Data Objects [Online] Available: http://access1.sun.com/jdo/

[27] FastObjects (2003) FastObjects t7 [Online] Available: http://www.fastobjects.com/us/FastObjects_PandS_t7.asp

SSRI *online*
First experiences in a three-years course degree offered in e-learning at the University of Milan (Italy)

Ernesto Damiani[1], Antonella Esposito[2], Maurizio Mariotti[3],
Pierangela Samarati[4], Daniela Scaccia[5], Nello Scarabottolo[6]

Abstract

This paper is aimed at presenting SSRI online: an e-learning initiative started at the University of Milan (Italy) for the academic year 2004/05. The initiative consisted in offering an already existing three-years academic degree course ("laure a in Sicurezza dei Sistemi e delle Reti Informatiche - SSRI": a post-bachelor course on security of computer systems and networks) not only in the "traditional" way (mainly based on classroom lectures and labs) but also online.
Main aspects discussed in the following sections are:

- *the players of the initiative and their roles;*
- *the adopted teaching model;*
- *the technological solutions identified and/or implemented to support the initiative;*
- *the characteristics of the student population choosing SSRI online;*
- *the first results obtained after almost one year.*

1. SSRI *online*: why and who

The University of Milan has a significant experience in the usage of technologies for supporting teaching.
Main player in this arena is an inter-departmental center (CTU) devoted to study and apply multimedia-based methodologies and technologies to teaching. During the last four years, CTU produced more than 300 e-learning modules complementing traditional lectures, and made available to students through a web platform (ARIEL) completely designed and implemented by CTU itself.
Previous experiences at the University of Milan had been carried out, mainly by the computer science and technology departments, using synchronous virtual classes to give lectures to remote students and making video recording of lectures to offer asynchronous support to students occasionally missing classroom activities.
However, all these previous experiences were designed to "support" traditional teaching, not intended as a "substitute" for it. Students working full time or living far away from Milan were supposed to integrate the e-learning material whit books, notes from colleagues, etc., without explicit help from the university.
Besides adopting a novel approach to e-learning, the SSRI course – launched in 2003 – is the first and unique three-years university course centered on ICT Security available in Italy. This uniqueness makes the course interesting for students spread over the overall country, and for this reason it became an ideal candidate for an e-learning version.
A further aspect making SSRI an ideal candidate for adopting e-learning was the fact that "traditional" version of it (i.e., the location were classes are given) is hosted at a campus located in Crema, a small town 40 km. South-east of Milan, where the Information Technology Department is situated. While the location provides a nice and quite environment for students going to the Campus, the geographical distance and limited public transportation make it not easily reachable. An online version of SSRI would have then be interesting also for students living in Milan but having difficulties in reaching Crema (no highway from Milan to Crema, time-consuming public transportations, traffic jam, …).
Based on the observation above, and willing to experience a initiative completely online, one year ago the University of Milan decided to activate SSRI *online*, involving in the initiative:

- the Department of Information Technologies in Crema, providing the courses and whose professors

[1] Full professor, responsible for online version of SSRI
[2] CTU director
[3] Full professor, president of the University commission for ICT strategies
[4] Full professor, president of the SSRI didactical coordination council
[5] CTU project manager for SSRI online
[6] Full professor, 1st year teacher in SSRI online

agreed to:
  - re-design their lectures for the online version;
  - identify and train tutors for online interaction with students;
  - overview online learning and make examinations;
- the inter-departmental center CTU, devoted to:
  - identify the best suited teaching model;
  - identify the technological tools allowing the production of the online version of SSRI;
  - drive the production of the online material, to ensure quality and homogeneity among the various courses;
  - implement and manage the platform offering SSRI *online*;
  - provide process tutoring (daily interaction with students, closely monitoring the student s' progresses, etc.);
  - handle – in conjunction with the student secretariat in Crema – all the logistics aspects.

To support the design of the didactical model and of the online material, the university team has been complemented by consultants from Isvor Knowledge System, a company specialized in the production of e-learning courses.

2. SSRI *online*: how

The structure of SSRI *online* can be summarized as follows:

- each online course is structured in modules (each completing a given topic of the course). Each module is composed of didactical units, associated with the various aspects of the topic and constituted by different activities: lectures, exercises, tests;
- all teaching material is available to students on the CTU platform, which provides also forum discussions among students and tutors;
- students' progresses are monitored by tracking their activities and the results of the tests associated with each online lecture;
- online activities are coupled with face to face meetings between students and professors. For each courses there are three face to face meetings at course start (course introduction), halfway in the teaching period (mid term exam), and at course end (final exam).

To discuss in further details the didactical organization of SSRI online, we can first stress the fact that the course has neither been intended as a simple distribution – through Internet – of learning materials, nor as the simple online offering of the "traditional" classroom courses. SSRI online implied in fact the complete re-design of the full didactical and organizational environment related to the classroom version of SSRI. It is then more appropriate to refer to SSRI online as an "online laurea course" instead of "the online version of a laurea course".

2.1 The didactical environment

To support the re-design of the didactical contents, the teachers have been supported by a group of Instructional Designers, coordinated by CTU. Such a group focused first on de-structuring each single course, to identify its autonomous parts (**Modules**) and their components (**Didactical Units**). In turn, each Didactical Unit has been organized in several **Lectures**, each with associated a number of **Activities** to be performed by the students.

This process lead to the identification of the learning objectives of each Module and each Didactical Unit, their formulation in terms of "knowledge" and "practicalities", their articulation in terms of learning moments (where students are expected to learn concepts) and operating moments (where students are expected to apply the learned concepts).

This macro-design phase has been followed by a micro-design phase, aimed at identifying, for each activity, the multimedia element more suited, from a didactical-methodological point of view, for the presentation of the activity contents.

Among the various multimedia elements adopted, the most important ones have been the following:

- video recording of the teacher, mostly used to present – at the Module level – the contents and the objectives of the Module itself. Even if not used in this first academic year of SSRI online, the same technique could obviously be used also to support critical point inside single Lectures;
- sequences of slides synchronized with the teacher's voice, used – at Activity level – to focus the attention of the student on key elements of the Activity, to give deeper insight into Activity concepts, to support the concepts with examples. The reason for choosing to avoid video recording has been not only to guarantee delivery time, but especially to make students concentrate on graphical/textual material instead of teacher's behavior;
- desktop capturing, synchronized with teacher's voice, to allow the teacher to explain the behavior of particular programs, show interesting websites, and help the students to become independent in the usage of the desktop itself;

- blackboard-like elements, where the teacher can simultaneously record her voice and her handwriting activity on the screen: this can be done either with an empty screen (thus reproducing the classical blackboard lecture) or with a previously prepared background (e.g., a graphic or a drawing) where the teacher can add and/or emphasize parts;
- textual/graphical lecture notes (in form of book chapters) to integrate online learning with the traditional offline reference tools.

The design of these different multimedia elements had the objective of supplying the teacher with a set of tools in order to better adapt the online learning material to the nature and the complexity of the single teaching activity.
As a result of the adoption of the above multimedia elements, each SSRI online lecture is then constituted by a video recording (10 to 15 minutes long) presenting visual material (slides, PC desktop activities, etc.) accompanied by the teacher's voice. To ease production and revision of the lectures, the following approach has been adopted by CTU:

- visual material is prepared by teachers following some guidelines (e.g., slide templates);
- lectures are recorded autonomously by teachers (whenever and wherever they want) using programs (SofTV Presenter and Camtasia) that allow the synchronization of desktop activities and teacher's voice;
- recording is done on tablet PCs, allowing also blackboard-like behavior by teachers (any handwritten note made by teachers on the tablet PC screen during recording becomes part of the lecture itself);
- post-production is limited to a consistency check of the final lecture and to some aesthetical interventions (e.g., smooth transaction between slides).

All the online learning material has been published on **Ariel.net**: the e-learning platform developed and implemented by CTU.

2.2 The Ariel.net platform

Ariel is the e-learning platform developed in the past few years by CTU to allocate the various e-learning materials developed by CTU itself to support web-enhanced university courses.
After a deep benchmarking activity, the main conception of Ariel proved to be suitable to support SSRI online, even if several additional functionalities were required to pass from web-enhanced to real online teaching. In particular, tutorship support and tracking of students learning were absolutely mandatory for a complete online platform, willing to support not only SSRI online, but also future similar courses at the Milan university (e.g., masters, specialization courses, etc.).
To this purpose CTU designed and implemented the **Ariel.net** platform, based on the Microsoft .net technology. In designing the platform, particular care has

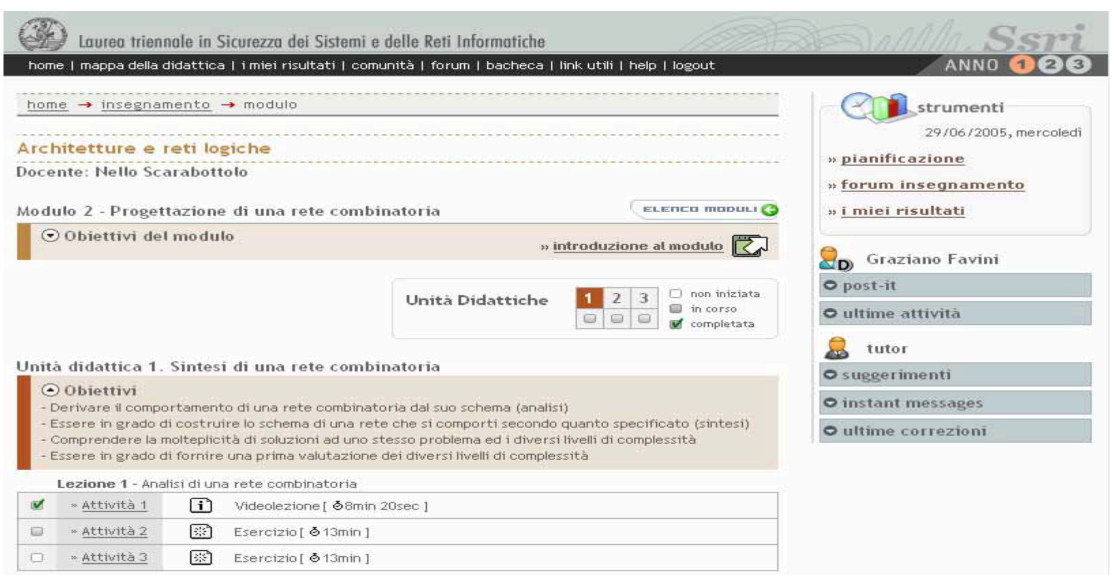


Figure 1

been put in the integration of the various tools, to allow context-driven "navigation" among them: for instance, a student following a specific course can easily pass from lectures of that course, to the forum of the course, to interaction with tutors of the course, without leaving the specific course itself. As it can be seen in Figure 1, the right part of the screen shows (in Italian…) the direct links a student has when she is following a given course: forum of the course, annotating tool, tracking, communication tools with the tutor. Whenever the student switches to another course, all links adapt themselves to the new situation.

The qualifying functionalities of Ariel.net allow:

- the support of one-to-one as well as one-to-many communications, both asynchronous and synchronous. Besides traditional e-mail and forums, Ariel.net supplies also a private messaging system among students and tutors integrated into each single didactical activity (instant messaging), a virtual bulleting board reserved to tutors to post general interest messages, a virtual classroom support for synchronous meetings among students and tutors/teachers;
- handing the access to courses by students, forcing them to adopt the quarter period structure (courses are accessible only under platform control);
- self-planning of learning activities by each student, who has a suggested learning plan, but who can change this plan according to her own needs. The plan is accessible by tutors, who can then track student work and intervene in case of evident pace loss. As in can be seen in Figure 2, for each Module of a given course, the temporal diagram shows three lines: the black, upper line is the study time proposed by the platform; the brown, intermediate line is the personal plan of the student; the blue, lower line is the actual progress of the student her/himself;
- both online streaming fruition of audio/video elements, as well as download for offline fruition;
- the support of the exercising phases of students, tracking their advance and their results. Exercises can be either multiple-choice, closed-answer tests (Ariel.net gives immediate feedback to the student), or tests whose correction is done by tutors (Ariel.net supports the sending of the student solution to the tutors) or tests whose correction is done by comparison with the solution proposed by the teacher;
- the ability to closely follow and support the individual learning process of each student, through a tool allowing each student to annotate her/his own instance of the online material (the annotation becomes a virtual sticker, "glued" to the material). For the tutor, this tool is used to make

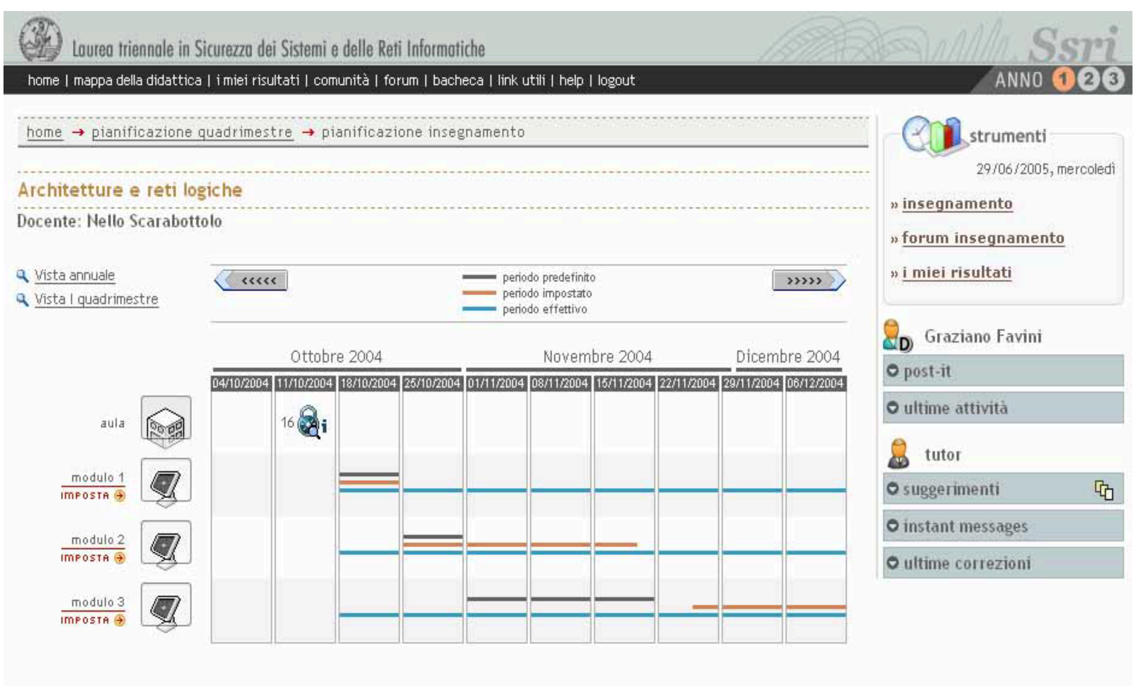


Figure 2

comments available to all students;

- handling of logistical aspects as subscription lists to intermediate tests and final exams, recording of obtained grades, etc.

A pictorial representation of the architecture of Arial.net is represented in Figure 3.

(different study times) and for psychological attitudes;

- frequently is a worker, who stopped her/his studies several years before coming to SSRI online, thus encountering particular difficulties in re-defining her/his own study pace.

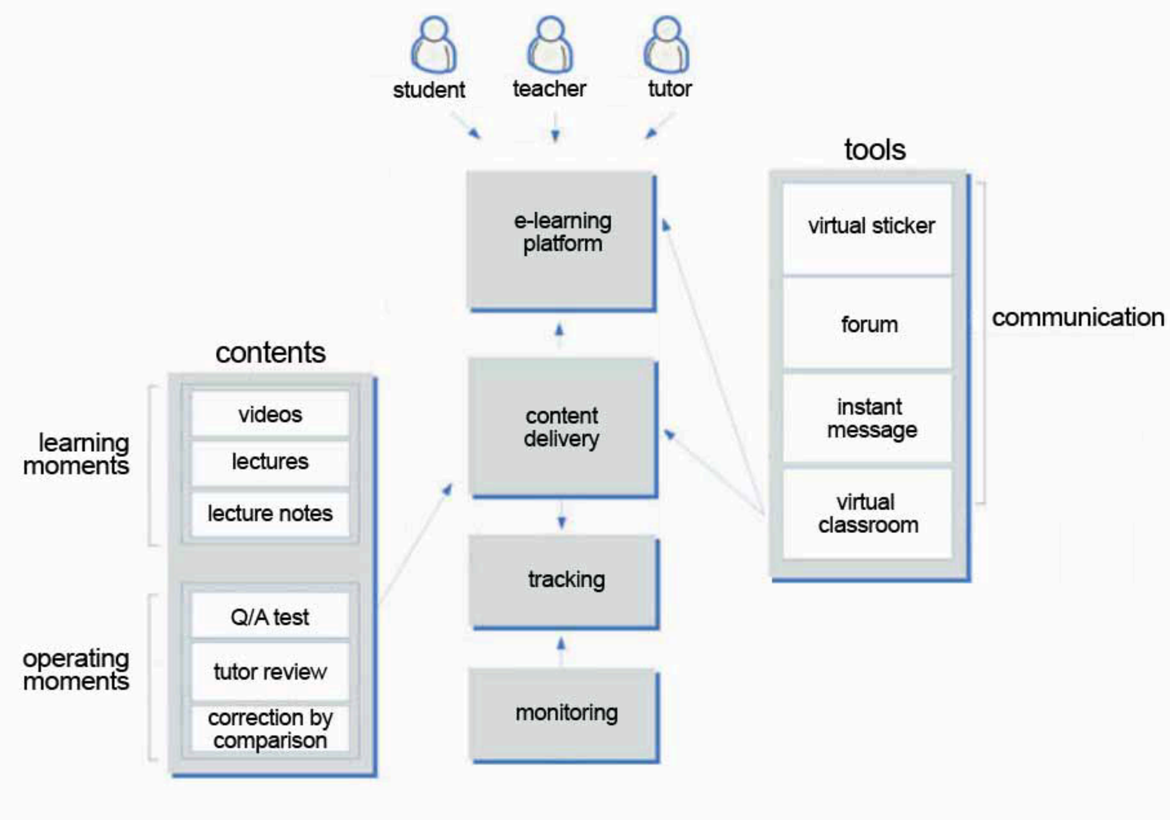


Figure 3

2.3 The organizational environment

Coming to the re-design of the organization of SSRI, the most critical aspect taken into account has been the peculiar target of potential users: in fact, the typical online student:

- is isolated both from the didactical context and from the colleagues following the same courses in the same period of time; the lack of interactions with teachers and colleagues makes particular important the possibility to immediately apply learned concepts to verify learning progresses;
- could have difficulties in being involved into collaborative works, both for logistical reasons

Taking into account the above aspects, SSRI online has been structured around a three four-month periods calendar, allowing the student to follow a reduced number of courses in each four-month period.

Each single course plans two intermediate evaluation tests, used not only for self-evaluation purposes, but also for integrating the final grade.

Reserved exams for online students have been planned at the end of each four-month period, before the starting of the next one, in order to clearly separate test times from learning times. Moreover, to help working students, tests and exams have been organized on Friday and Saturday.

3. SSRI *online*: to whom

SSRI *online* has been opened in the first year to a maximum number of 120 fresh students, on a first-come-first-accepted basis: the number has been saturated in just two weeks, and at least 100 more students have declared their interest in the online approach.

It is worth noting that this success of SSRI online did not affect the number of subscription to the traditional version of SSRI, which received 138 freshmen.

A deep analysis of the population of SSRI online students is beyond the scope of this paper: however, it is worth summarizing the most important outcomes:

- the large majority of the students is constituted by people older than traditional freshmen (67% of SSRI students are more than 28 years old, 23% are more than 35 years old) and already working (95% of them are full-time workers). This means that the online approach does not constitute a real alternative to the traditional one, since it definitely addresses a different student population;
- looking at the geographical distribution of SSRI students, it appears that 77% of them live and work in Northern Italy (where the University of Milan is located) 15% in Central Italy and only 8% in Southern Italy. This seems to drive to the consideration that physical distance is not as critical as time constraints in the decision of selecting an online course.

To further investigate into student motivations, two questionnaires have been distributed, one at the beginning of the first four-month period (to ask about expectations) and a second at the end of this period (to ask about fulfillments of the expectations). Main results are the following:

- the large majority of students (more than 90%) declared that, without the online opportunity, their would never have subscribed to a university course. This clearly shows that the online approach, far from being a fashion, satisfies a real need for people already working and willing to improve – at their own pace – their professional capabilities;
- among the main motivations for selecting an online course, the most frequent is the flexibility in study time organization, followed by the opportunity of a professional growth;
- even if the time flexibility is the main expectation, several students (26%) also expected the online approach to be more effective than the traditional one, 58% of them expected to reach the same level of competence of traditional students, and 17% of them expected an even deeper competence. This seems to indicate that the personalized interaction guaranteed by tutors is perceived as a significant improvement towards traditional classroom-based learning;
- the expected quality of the learning process is confirmed at the end of the first four-month period, when 59% of the students consider it very effective, 36% of them effective, and only 5% of them not effective enough;
- also the delivery platform Ariel.net is well perceived: most of the students declared to be able to easily access and download the online material, whose quality and clarity have been deeply appreciated.

Even the learning results of online students – compared with the results of the students following the traditional SSRI courses – reveal the correctness of the approach we adopted: both percentages of students passing the examinations and average grades obtained are aligned between the two populations.

4. Concluding remarks

The online course presented in this paper is presently entering the third and last four-month period of the first year, and a complete picture of its effectiveness has to be necessarily postponed to the end of such a first year.

Nevertheless, the preliminary results obtained from several information sources (personal data of SSRI online students, questionnaires, exam results) definitely show the fact that an online learning approach fills a real gap left by traditional classroom teaching, and that a course completely re-designed to be offered online is well appreciated by students and leads to learning results absolutely in line with traditional ones.

These preliminary results convinced the board of the Milan University to extend to 200 freshmen the access to SSRI online for next academic year, and to organize an admission test to select the most motivated and prepared students.

A Web-Based Architecture for tracking Multimedia using SCORM

P. Casillo, C.Cesarano, A. Chianese, V. Moscato
Dipartimento di Informatica e Sistemistica, University of Naples Federico II, via Claudio 21
80125 Naples, Italy
{p.casillo,carmine.cesarano,angchian,vmoscato}@unina.it

Abstract

In the context of a driven and collaborative on-line learning, multimedia resources, in particular videos, are more and more used. Due to their complex nature, there is the increasing need to manage the video contents in order to ensure a more fine-grained tracking on audio-video assets and obtain a continual feedback of student activities. In according to the ADL's SCORM model, a video is usually assumed to be an atomic Learning Object: this assumption restricts the interaction between client and Learning Management System to a merely ON/OFF tracking process and limits the reusability of such type of contents. In our approach the video resources are considered as SCO or, more in details, as a SCO container; in this manner, each single component asset can be just a frame or a video segment in order to achieve the wanted tracking grain size. In according to the last hypothesis the main focus of the paper has been the design and implementation of an architecture for the tracking of video SCOs in web learning environments.

1 Introduction

Usually learning is seen as involving three processes[8]:

- **acquiring skills, constructing knowledge or concepts, and developing values**: these correspond roughly with the use of technologies (understood not only as involving computer-based technologies but a wide range of tools including reading and writing);
- **the use of knowledge resources**;
- **the negotiation of shared meanings and values** among the members of a learning community.

In e-learning environment such processes take place specifically involving computer-based tools, accessing digital knowledge resources, and participating in computer-mediated or online communities. Numerous authors have discussed the benefits of ICT course delivery for learners, tutors and institutions. Technology has been used as a means of electronically distributing course material, allowing flexibility for students favored learning styles, pace, etc. and giving greater access to information; as well as enabling remote communication between students and tutors, and between student peer groups. On the course management side, it can allow greater communication with the course team and provides flexibility to maintain and update course material and documentation [7].

To this purpose [1], multimedia resources can yield a more contribution to learning process especially in e-learning scenario, in which the web is the front-end of the contents presentation. In this particular context, the videos are generally considered the most diffused resources for multimedia presentation and they are assumed as atomic objects with a set of global metadata. This assumption restricts the interaction between client and Learning Management System to a merely ON/OFF tracking process and limits the reusability of such type of contents. The problem is to characterize the single parts of the video that can be utilized for its traceability: for example, it should be useful divide the video in scenes and control their learning because these video segments can be considered as part of a learning process. In our approach the video resources are considered as SCO or, more in details, as a SCO container. In our approach the video resources are considered as SCO or, more in details, as a SCO container. In this manner, each single component asset can be just a frame or a video segments in order to achieve the wanted tracking grain size. In according to the last hypothesis the main focus of the paper has been the design and implementation of an architecture for the tracking of video SCOs in web learning environments.

2 Background and Motivations

The SCORM (Sharable Content Object Reference Model) metadata Information Model is a reference to the IMS Learning Resource metadata Information Model, which itself is based on the IEEE 1484.12.1 LOM (Learning Object metadata) standard. The SCORM [2], [3] defines a Web-based learning "Content Aggregation Model" and "Run-time Environment" for learning objects.

The purpose of the SCORM Content Aggregation Model is to provide a common means for composing learning content form discoverable, reusable, sharable and interoperable sources. The SCORM Content Aggregation Model further defines how learning content can be identified and described, aggregated into a course or portion of a course and moved between systems that may include Learning Management Systems (LMS)and repositories. The purpose of the SCORM Run-time Environment is to provide a means for inter-operability between Sharable Content Object-based learning content and LMSs. A requirement of the SCORM is that learning content be inter-operable across multiple LMSs regardless of the tools used to create the content. For this to be possible, there must be a common way to start content, a common way for content to communicate with an LMS and predefined data elements that are exchanged between an LMS and content during its execution.

Content Model is a nomenclature defining the content components of a learning experience. Content Packaging defines how to represent the intended behavior of a learning experience (Content Structure) and how to package learning resources for movement between different environments (Content Packaging). The SCORM Content Model Components is composed of three components:

- **Asset**
- **SCO**
- **Content Aggregation** between the members of a learning community

A SCORM **Asset** is a collection of one or more resources (Whole web page, HTML Fragment, JavaScript Functions, Flash Object, JPEG Images...) that are appropriate for sharing among SCOs; when packaged, an Asset should contain the appropriate metadata making it searchable in a SCORM repository. Sharable Resource metadata is data contain in the resources manifest file that describes the resource. An Asset can be described with Asset metadata. An Asset metadata is a definition of metadata that can be applied to "raw media" Assets that provides descriptive information about the Asset independent of any usage or potential usage within courseware content. This metadata is used to facilitate reuse and discoverability principally during content creation, of such Assets within, for example, a content repository. Assets are not only resources that can be shared among SCOs, they are also resources that have the capability of being discovered in SCORM repositories.

SCOs, Sharable Content Objects, are built using resources, such as web pages, graphics files, etc. In some cases, these resources might need to be shared with other SCOs. Resources that can be shared with other SCOs are called Course Assets, or Sharable Resources. According with the SCORM documentation, a SCO essentially consists of three defining characteristics:

- a SCO is the lowest level component that might be used in another course;
- a SCO should provide useful learning content by itself;
- a SCO must be designed to be launched and tracked by a SCORM-compliant LMS.

In other words a SCO is a set of related resources that comprise a complete unit of learning content compatible with SCORM run-time requirements. With this definition, a SCO can be extracted from a learning object and used by another learning object. This is essential to achieving the SCORM reusability goal. A SCO represents a collection of one or more Assets that include a specific launchable asset that utilizes the SCORM Run-Time Environment to communicate with Learning Management System (LMSs). A SCO represents the lowest level or granularity of learning resources that can be tracked by an LMS using the SCORM Run-Time Environment. A SCO is required to adhere to the SCORM Run-Time Environment. This implies that it must have a means to locate an LMSs API Adapter and must contain the minimum API calls (LMSInitialize() and LMSFinish()). There is no obligation to implement any of the other API calls as those are optional and depend upon the nature of the content.

A **Content Aggregation** is a map (content structure) that can be used to aggregate learning resources into a cohesive unit of instruction (e.g. course, chapter, module, etc.), apply structure and associate learning taxonomies. In SCORM *A Learning Object, or SCORM Content Aggregation, is a collection of Sharable Content Objects (SCOs) described by a SCORM manifest file*.

3 System architecture

3.1 Design Objectives

The main goal of this work is the design and implementation of an architecture that ensures the following features in the e-Learning process:

- **Adaptivity**
- **Interactivity**
- **Openness**

In the following we describe more in details the listed features.

Adaptivity is needed to select and customize the learning resources to the learner/student and to the context in which the learning is taking place. These two aspects exhibit a wide range of variability for an application that is web-based. Such application can not make a-priori assumptions about the characteristics of the learner, such as educational background, cognitive style, etc., nor about the context and purpose for the learning process. Instead it must be able to adapt dynamically based on explicit knowledge about these aspects that need to be maintained independently of the more generic learning content knowledge.

Interactivity is desirable to make the learning resources more responsive, autonomous and proactive, and to better exploit the inherently distributed nature of web-based e-Learning. Web-based learning lacks the advantages of the traditional student-tutor relationship as it involves learners interacting on their own with the e-Learning application. Such application needs to be able to approximate many of the characteristics of a human tutor or coach, such as being non-obtrusive, have a good feel or sensing of the status of the student with respect to the subject being learned and be able to adopt motivational strategies among others.

Openness is a requirement for any technology that aspires to become a global undertaking spanning many collaborating institutions, within many cultures, and across many languages. Experience has demonstrated that technology alone is not sufficient to achieve successful adoption of new solutions. Open standards have shown to be a key factor in the achievement of widespread adoption, by facilitating re-use, specialization, and quality improvements.

3.2 Architectural Layout

All the above objectives point in the direction of a web-based solution. Among the actual web technologies, we have chosen a **mixed Browser-based/Client-based implementation**. To communicate with LMS the learner has only to support javascript (as in ADL's SCORM recommendation); the other active portion at the learner is a plug-in which permit the media to be transmitted to the learner and facilitates the interaction with him. The reliance on the Browser although allowing the plug-in to be a relatively thin client, however imposes/inherits all the Browser limitations upon the plug-in, constraining the ultimate flexibility of the approach.

For what concerns the server side the proposed architecture (see figure 1) for video tracking is made up of different modules: some of them are parts of typical web-based learning environment, other ones are the components of a classical video management architecture [6]. A *glue* infrastructure has been introduced to enhance the tracking process and fully support trackable video streams.

Most of the components of the shown diagram are also common parts of commercial industry-standard LCMS and LMS products and different background colors remark that a block belongs to one of these categories [5]. As shown in figure 1, the proposed architectural layout is a set of five different entities:

- **Lightweight Directory Access Protocol**: the whole system relies on a common LDAP authentication infrastructure to provide efficient and affordable access control to the courses and a per-user video navigation-level tracking.
- **Learning Management System** which stores the *learning contents* provided by the *content experts* by the means of *learning content authoring tools*. The system receives the course contents from the LMS architecture and delivers courses through a pool of content servers also denoted as *Delivery Servers*. Its other main finalities are to store per-user profiling information, course details and manage the course authoring phase. Progress-tracking information are also retrieved and stored by the tracking engine which controls the content delivery.
- **Learning Content Management System** which stores the *learning resources* provided from teachers, tutors and content experts. It is a support for improve the functionalities of the Traditionally LMS. It enhances the traditional process of sharing learning resources through all the actors/actresses of the System.
- **Learning Authoring Tool**. It's a Client/Server application which allows the Content Experts to produce Learning Objects according to the SCORM standards. The Learning Authoring Tool allows to assemble the Learning Object and Asset to create SCOs and provide simple instruments to create a structured course. Then it can communicates with the LMS to deploy the package in which the course (and its metadata) is stored.
- **Streaming Server**. It's the component of the architecture in which the Multimedia files are stored; its work is to the deliver the Multimedia files to the Learner and / or (during the on-line management) to the Teacher.
- **Multimedia Client**. It's a Client/Server application which allows the Content Experts to catalogue, manage and logically divide the Multimedia Objects. It also allows the Content Experts to transform the Multimedia Asset in a Multimedia SCO.
- **Browser and Plug - In**. No more than a traditional browser: for the Learner, it has only to allow him to see the Multimedia files and communicate whith the LMS through javascript; for the Content Experts and

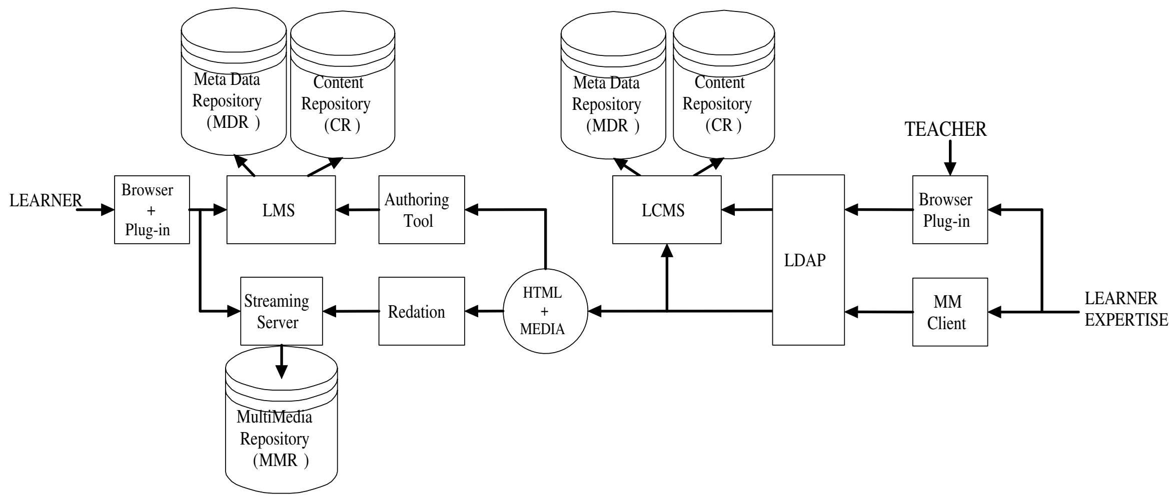


Figure 1. System Architecture - a *glance*

the Teachers, it allows to change on-line the description of the Multimedia Object.

Most of the items of our layout are conform to the general implementable e-Learning architectures described by known literature as in [5] and [6]; and most of these items are commercial systems (see 5 *Current Implementation*). The remaining parts (as the LCMS) are indeed handcrafted to support *deep* Multimedia content tracking and driven content navigation. In the next section we deeply care about the Multimedia Client which is the main focus of our work.

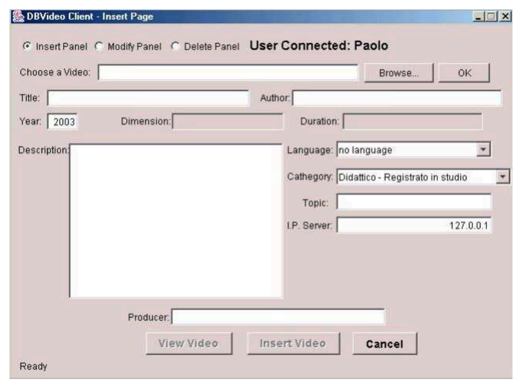


Figure 2. Multimedia Client - general metadata

4 A Client for Multimedia Learning Content Management

The design and development of a client application able to manage multimedia resources with didactic aims is a current research challenge in the LCMS field. From a general point of view, such client has to be considered an integrated system capable, from one hand, of satisfying, all learner requests, from the other one, of simplifying and accelerating the content-expert metadatation task (Figure 2).

In order to achieve these purposes the didactic video is at first segmented into shots in according to a well-know automatic indexing process [4]. After that, the shots are grouped into scenes, that can be considered as video parts having a particular semantic meaning, by means of a semi-automatic process (Silence Detection) [9] driven and supervised by the content-expert. In this manner the video is not assumed as a closed black box (autonomous LO) in the learning process but, it can be seen as a SCO container where each SCO can be a video scene.

As underlined previously a crucial focus is to choice the algorithms for video indexing. In particular, for video shot segmentation we have used a technique based on Animate Video [4] in which video cuts are determined on the base of local minimum points detection thought a visual attention function to make this logical partition. On the opposite, for video scene detection in literature we have a lot of different algorithms which could be chosen for this purpose [11], [12]. Our study has been addressed by two different needs: (i) we have to manage ”learning” multimedia file which are, in most of cases, very poor in shot contents; (ii) we have to use software which really improves and simplifies the work of the content expert. For these reasons, in the context of learning-video, scenes can be correctly detected by a Fast Silent Detection[9].

The next step is the metadatation of the video scenes and a correction (usually making a manual video scenes merging) of the automatic algorithm results that, in the majority of cases, can be caused false detections (Figure 3).

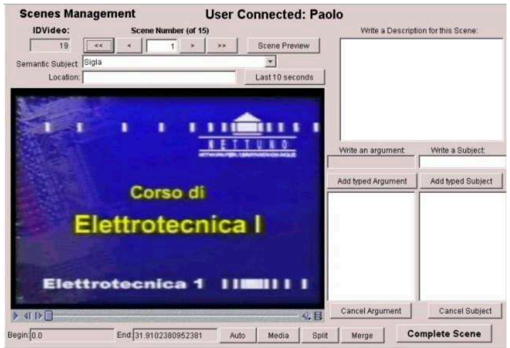


Figure 3. Multimedia Client - scene metadata

Using this approach it is possible to build up an annotated video database in which each multimedia content (consequently each scene) can be indexed and retrieved by a dedicated search engine, in this manner, teachers can use it to enrich their courses. All metadata are stored in the LCMS and can be used by the LMS to dynamically offer a detailed description of multimedia contents.

For what concerns the SCORM tracking process we assume the following hypothesis:

- What is to be tracked is always a *HTML page*
- This *HTML page* is linked to a LMS by invoking appropriate API
- These invoking procedures contain information about: (i) the resource (page) to be tracked, (ii) a flag indicating if the resource has been already seen by the learner, (iii) the time (expressed in seconds) spent for the resource fruition and the user break-point, (iiii) a further flag indicating if the fruition is completed.

In particular, to invoke the communication API and manage the data exchange with LMS, for each SCO some javascripts are opportunely included in a HTML page (by inserting appropriate video starting/stopping buttons, see Figure 4; for this reason the control buttons of video player are disabled) that drives the learner fruition on the resource.

In the description bar of video player information about SCO, picked up on-fly by the LCMS in which they are stored, continuously appear. By the pressure of stop button or in correspondence of the video end, the fruition state of the SCO is updated. In this way the teacher can be always informed on the progress of learner and monitor students activities .

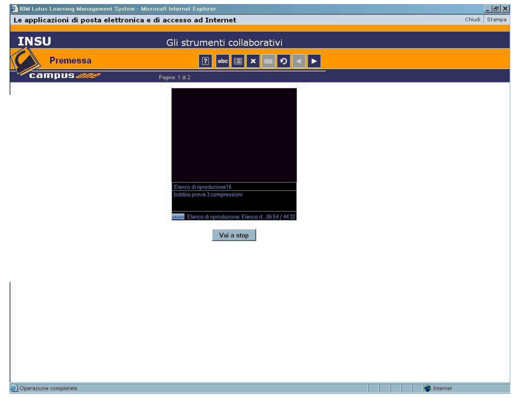


Figure 4. The HTML page for Multimedia Content fruition

5 Current Implementation

The proposed Multimedia tracking system, developed in the e-learning framework of University of Naples ”Federico II”, operates in according to the following specifications.

- The videos are converted using the *Windows Media Encoder 9 Series* encoders to publish video assets by the means of the *Microsoft Windows Media Services* streaming server provided in the standard *Microsoft Windows Server 2003* installation. This product has been chosen because of the advanced functionalities of bandwidth-adaptive streaming. We made successful experimentations also with Darwin Streaming Server.
- The core element of the system is the *IBM Websphere 5.0* application server which deploys a customization of the *IBM Learning Management System 1.0.5*. But we have succsessfuly tried this architecture on Oracle iLearning 4.x.
- The LMS application uses a common authentication infrastructure and a uniform naming schema based on the *Oracle Internet Directory 9.0.2* LDAP server.
- The ’timeline’ dependencies and metadata into the Multimedia-SCO are managed by a J2EE servlets deployed on our running instance of the *Tomcat 5.0.x* application server. This application relies on the underlying database tier to store and compute info about the time intervals that the students spend browsing each single video element of the course.
- The Multimedia Client has been developed in Java and is compatible with JVM 1.4 or sup.. JavaTM has been selected as reference platform mainly for its characteristic of portability.

- The standard tracking info and the additional timeline dependencies are stored in schemas provided by a large *Oracle 9i* database instance. Platform metadata and per-user tracking data are separated by our additional time data by the means of additional database schemas.

The streaming server runs on *Microsoft Windows 2003 Server*. The remaining parts of our architecture runs on a *Linux Red Hat Advanced Server 2.1* distribution.

Eventually additional application server is a set of Tomcat instances running on *Debian GNU/Linux 3.0r2* distribution customized with recompiled critical-mission software infrastructural components. We merged three different video compression levels into a single video stream to adapt the performances to the standard baudrates of the common PSTN, DSL and LAN/Fiber connections.

6 Conclusions

A web based system for video LO tracking has been presented. It allows to users a driven browsing on video contents in a e-learning environment and to LMS the possibility of maintaining information about user progress status on such educational resources. Future works will be devoted to three different study directions:

- Create a *MultiMedia mark - up language* for learning objects which will allow to optimize the interaction between the Multimedia Client and the LCMS [10]. At this implementation, for each change in multimedia description, the MMClient communicates with the LCMS DB so the Content Expert has to be on line to use this software. In future release, the Multimedia Client will produce an XML as result which had to be uploaded on the LCMS; in this way, all the changes will become effective in one time.
- *The LCMS will produce the HTML pages*: in this way the teacher will not need to install the Multimedia Client anymore. And he will be able to obtain all that he needs for multimedia contents every time and everywhere on line, without direct interaction with Content Expert nor any software (except Authoring Tool).
- Our efforts are concentrated to create a *software which will computerize the search of ontologies in the multimedia content*.

7 Acknowledgments

This work has been carried out partially under the financial support of the Ministero dell' Istruzione, dell' Universita' e della Ricerca (MIUR) in the framework of the FIRB Project Middleware for advanced services over large-scale, wired-wireless distributed systems (WEB-MINDS).

References

[1] A. Large,J. Beheshti, A. Breuleux, A. Renaud, "THE INFLUENCE OF MULTIMEDIA ON LEARNING: A COGNITIVE STUDY", Proceedings of the ACM Multimedia '94, pages 315–320, San Fransisco, CA, October 1994.

[2] www.adlnet.org

[3] "Shareable content object reference model initiative (SCORM), the xml cover pages", October 2001, http://xml.coverpages.org/scorm.html.

[4] G. Boccignone, A. Chianese, V. Moscato, A. Picariello, "Foveated Shot Detection for Video Segmentation", IEEE Transaction on Circuits and Systems for Video Technology, VOL. 15, NO. 3, MARCH 2005.

[5] "E-Learning Applications Infrastructure", White Paper, SUN.

[6] S. Shermann, M. Chang, Q. Li, "Architecture and Mechanisms of a Web-Based Video Data Management System", Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'00), New York City, NY, USA, Jul 30-Aug 2, 2000.

[7] eLearning Industries Group, "Developing eLearning Communities in the EU - The eLearning Industry Group Perspective".

[8] H. Beetham,"Understanding e-Learning", Skills for e-learning, Ulster, 29 July, 2002.

[9] P. De Souza, "A statistical approach to the design of an adaptive self-normalizing silence detector", Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing], IEEE Transactions on Volume 31, Issue 3, Jun 1983, Page(s):678 - 684.

[10] G. Amato, C. Gennaro, P. Savino, F. Rabitti. "Milos: a Multimedia Content Management System for Digital Library Applications", Proceedings of the 8th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2004), Volume 3232 of Lecture Notes in Computer Science, pages 14-25. Springer, September 2004.

[11] A. Hampapur, R. Jain, and T. Weymouth, "Digital video segmentation, Proc. ACM Multimedia94, 1994, pp. 357364.

[12] H. Lu, "Model based video segmentation, IEEE Trans. Circuits Syst. Video Technol., vol. 5, no. 5, pp. 533544, May 1995.

A GQM Based E-Learning Platform Evaluation

B.Fadini(+) P.Maresca(+) P.Prinetto(*) C.Sanghez(+) G.Santiano(*)

(+)
Dip. di Informatica e Sistemistica
Un. di Napoli "Federico II" - Naples, Italy
E-mail {fadini, paomares, sanghez}@unina.it

(*)
Dip. di Automatica e Informatica
Politecnico di Torino - Turin, Italy
E-mail paolo.prinetto@polito.it

Abstract

Several E-learning platforms are today available, both open-source and proprietary.

When a platform has to chosen, a detailed evaluation is needed.

Evaluation E-learning platforms requires evaluating non only the implementing software package, but additional features as well, including, among the others, the supported teaching and delivering schema, the provided QOS, the global dependability, and the guaranteed scalability.

The paper presents a paradigm to evaluate E-learning platforms based on the Goal/Question/Metrics (GQM) methodology [□, □]. The proposed approach is general enough to allow performing detailed comparative analysis of different platforms.

1 Introduction

Distance Learning and, more specifically e-learning, have nowadays reached their third generation. Several enabling factors played key role in today developments, including, among the other, the wide acceptance of the concept of Learning Objects [3], the availability of a plenty of open source e-learning platforms, and the diffusion of standards, like SCORM [4], to improve interoperability.

When a new customized platform is need and several criteria lead to decision of adopting and customizing an open source available platform, designers have to face the problem of finding a solution that best fits target constraints.

The present paper presents a methodology the authors adopted to systematically evaluate e-learning platforms, and is structured as follows. In section 2 we show the extension of GQM process to e-learning platform. In section 3 we show the application of GQM process to e-learning platform. In the section 4 we discuss the results and in the section 5 we state the conclusions.

2 The adopted methodology

The main goal of the research activity outlined in the present papers was the "evaluation" of the quality of an e-learning platform w.r.t. some given application constraints.

When approaching the problem, we realized that, eventually we had to evaluate a "software product", by performing a significant campaign of measures.

If, from the one hand, no accepted standards are available for planning a data gathering process, from the other hand Vic Basili proposed the so called *Goal/Question/ Metric* (GQM), a significant paradigm for Software modeling and measurement [1] [2], whose basic steps are summarized in Fig. 1.

A deep feasibility study, suggested us to adopt the proposed process. The most significant motivations included, among the others, the following ones:

- The quality factors of an e-learning platform must be measured and monitored through the overall system life-cycle, including in-field and on-line activities of QOS monitoring
- The usage level of the target platform, in terms of # of users, was typically high, with some significant peaks and the QOS to be guaranteed may depend on the set of

actual users

- The number of attributes to evaluate and to monitor is so huge that a systematic approach to set up the process measure is definitely mandatory
- The *Quality Focuses* for an e-learning platform must be changed according to the different classes of users (typical examples being the evaluation of *usability* and *accessibility*)
- The measure process must be flexible and powerful enough to support the analysis of different platforms used to implement different teaching organizations and methodologies, and to track the "natural" evolution of a target platform in terms of new releases and upgrades
- The first phases GQM top-down approach (*Prestudy* and *Identification of GQM Goals* in Fig. 1) force the evaluators to carefully identify the set of the ultimate goals, thus preventing the error prone phase of extracting a meaning from a huge amount of data collected during the platform normal operations
- The GQM process enables evaluators to "slice" the target platform, in terms, for instance, of composing modules or available facilities, thus allowing them to comparatively cross-evaluating facilities through several platforms
- The GQM process allows a detailed analysis of the delivered teaching process, as well, thus minimizing the overall evaluation cost, since a single evaluation team is needed
- The defined metrics and the gathered data can be easily reused in further evaluation sessions
- Last but not least, the GQM methodology is particularly well suited to support and to implement the working methodology that the two research teams, cooperating on this activity, initially agreed on.

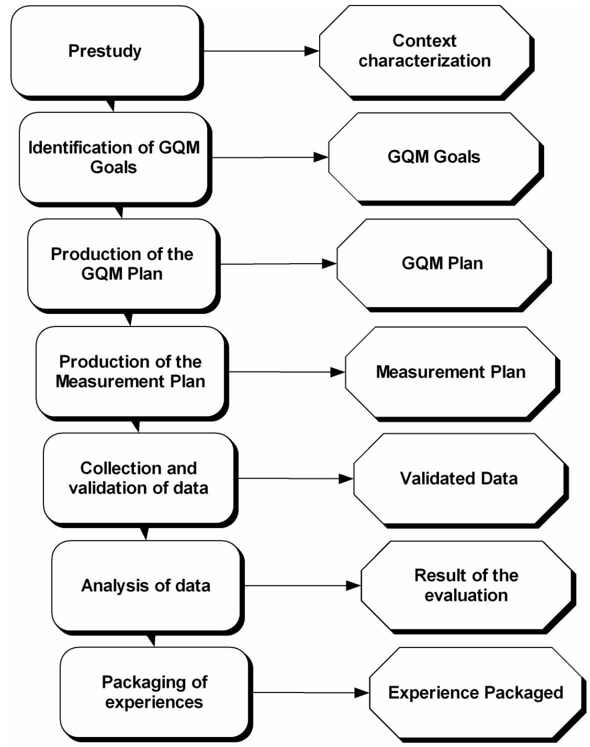


Figure 1 - The GQM Process

On the basis of the above considerations, we developed a comprehensive customization of the GQM methodology to evaluate any kind of e-learning platforms.
The proposed approach has been profitably used to perform a detailed comparative analysis of several open-source platforms.

3 The GQM process for e-learning platforms

In order to be applied e-learning platforms, the "generic" GQM process sketched in Fig. 1, must be carefully customized, taking into account the peculiarities of the target software packages, in terms of implementation details, developed functionalities, and ultimate performances.
For sake of generality, we first defined and detailed a GQM process general enough to be adopted in the evaluation of any e-learning platform. In a second phase, ad-hoc and customized focuses specific for each target platform have been identified.
In the sequel of the paragraph we shall focus on the major steps of process, as in Fig. 1, outlined

how they have been re-visited and adapted to our mission.

3.2 Prestudy

The *prestudy* phase for an e-learning platform aims at identifying the contest in which it will operate, with particular emphasis on the technical and implementation constraints imposed to the developers, the industrial strategic goals of the producer, the business models behind its development and its adoption.
For instance, it is important to identify:

- The target market (e.g., University, College, School, Public Company, Private company, any combination of all of them,...)
- Who is going to use it (e.g., On-line Service Provider, Continuous Education Provider, University, …)
- The delivered courses include tutoring and a tutor monitoring is required
- Legacy pre-SCORM [2] courses have to be uploaded and delivered
- …

In this phase all the available documentation in terms of maintenance reports, technical manuals, and user manuals is collected and analyzed. In addition several interviews are set up with designers, developers, platform managers, users to gather feedback and info that allow the evaluation team to understand the "external context" to the maximum extent.
Significant brainstorming and brain writing activities are needed in this phase.

3.3 Goal identification and hierarchical organization

While tackling the *Goal identification* phase, we soon realized that the cardinality of goals was so huge to force us to look for a hierarchical organization. The concept of *clusters of goals* has thus introduced.
The following clusters have been identified in this phase:

- *ISO 9126 Quality Characteristics.* Characteristic features of the platform seen as a software product (e.g., documentation, correctness, ease-of-use, ...) [5,6,7]
- *Website General Characteristics.* Characteristic features of the platform seen as an on-line accessible product (e.g., accessibility, scalability in term of number of concurrent customers, interoperability with different web browsers, ...)
- *Installation: basic hardware and software characteristics.*
  Characteristic features of the platform seen as a collection of functional modules (e.g., capability of inter-operating with different application servers, different DBMS, etc.)
- *Installation: basic hardware and software characteristics.*
  Characteristic features of the platform seen as a server to be installed by the customer (when applicable)
- *Standard Compliance.* Adherence to the different versions of the SCORM standard [4]
- *not SCORM-compatible.* Supporting the delivering of legacy courses,
- *Communication tools.* Functionalities provided by both asynchronous and synchronous communication tools (e.g., e-mail, forum, chats, messengers discussion groups, groupware functionalities ...)
- *Auxiliary learning tools.* Capabilities in supporting auxiliary teaching aids, (e.g., on-going self-evaluation test, hands-on experiences, simulations, ...) platform capabilities in supporting different teaching models (e.g., individual vs. class-based teaching, small vs. large classes, presence or absence of tutoring, several kind of tutors, …)
- *Course catalog and user management.* Characteristic features of the platform from the point of view of the system administrator (e.g., facilities for profiling and monitoring both single users and groups of users, facilities for identification and authentication, tracking and reports …)

- *Content management.* Characteristic features of the platform from the point of view of content management and authoring
- *Course catalog and user management.* Characteristic features of the platform from the point of view of the learning management system

For each cluster we identified several goals, globally obtaining 48 goals.
Goals have then been ranked according to their *significance*, resorting to the *Goal Selection Sheet* shown in Fig. 2 [8].
In such a sorting phase, a significant role is played by the info gathered during the *Prestudy* phase. As an example, the importance of the set of goals related to the SCORM standard compliance is strictly related to whether the target platform will be used to deliver learning objects developed according to that standard, or not.
Additional evaluation criteria include the impact that each goal can have of the other goals and the mutual *functional dependency* among goals. As an example, if the target platform has to be installed by the customer, the set of goals related to the platform installability have to be evaluated first.

	Urgent	**Not Urgent**
Important	Crisis pressing problems expired tasks	Preparation Prevention Development new activities Plan Expansion
Not Important	Some activities Useful activities in other contexts	Triviality Pleasant activities

Figure 2 - Goal Selection Sheet

3.4 The *GQM Plan* generation

As shown in the previous section, we identified specific reference goals for each cluster. These goals can be seen as “quality focuses” for the clusters and their generation has been made easy by the adoption of the Goal Abstraction sheet template shown in Fig. 3.
In this phase, these templates drove the brainstorming activities: thanks to the hierarchical approach we adopted, identifying the goals for the clusters proved to be functionally equivalent to determine the quality focuses in the classical GQM methodology.
Once goals have been identified, we reiterated the process and, resorting again to the Goal Abstraction sheet template shown in Fig. 3, we actually identified, for each goal, its quality focuses and the related key features.
The ultimate result of this step has been the generation of a working document, the GQM plan for e-learning platform, which has been profitably used as a guideline for the actual evaluation of several target platforms.
It worth pointing out that while implementing the adopted hierarchical iterative process we had to face some problems posed by the difficulties in finding unique and unambiguous associations between goals and clusters. To skip this kind of problems we are currently investigating two avenues of attack: from the one hand we are exploring the introduction of some additional levels in the hierarchy and, from the other hand, we are replacing the current tree structure by a directed acyclic graph structure.

Object	**Purpose**	**Quality Focus**	**Viewpoint**	**Environment**

Quality focus	**Variation Factor**
□ow can be described in detail the interest quality focuses?	Which factors can influence the interest quality focuses?
baseline Hypothesis	**Impact of Baseline Hypothesis**
Which are the values assumed by the interest quality focuses?	□ow the variation factors make to vary the quality focuses?

Figure 3 – The Goal Abstraction sheet template

3.5 Metrics definition

To best adhere to our hierarchical approach, we split the *production of measurement plan* phase into two sub-steps. The former one aimed at associating each quality focus with a specific metric; the latter one mainly consisted in the actual *measurement plan*, peculiar and properly customized for each target platform.

3.6 Measurement Plan: data collection, validation and analysis

As previously mentioned, in our approach the measurement plan is peculiar and customized for each target platform under evaluation.
According to the plan, the actual measurement campaign is then performed; data are gathered, analyzed, and properly validated. Data are eventually processed to generate the views, statistics, charts, and tables needed both to evaluate a single platform and to allow a comparison among different platforms.

4 Experimental results

Experiences are so far very satisfactory, mainly thanks to the easiness of the in-the-field data gathering campaigns, regardless the intrinsic complexity of the target problem. The actual GQM plan for an e-learning platform includes [10] so far 48 goals, 79 quality focuses, 495 questions, and 91 metrics.
The actual comparative evaluation of a set of open-source e-learning platforms is currently on the way both at the University "Federico II" in □aples and at Politecnico di Torino in Turin, Italy.
□ext activities will include a comparative analysis of the platforms currently under evaluation.

5 Conclusions

In the present paper we introduced a methodology to evaluate e-learning platforms, resorting to a customization of the well known quantitative approach GQM.
Such approach is sometimes criticized due to the *creative* aspects associated with the goal definition phase. According to our experience, the GQM proved to be a valuable methodology to define a clear set of goals, from which derive quality focuses, subsequently analyzed via data gathering campaigns and their interpretations via proper metrics [9].
In particular, we proposed a customized version of the methodology in Fig. 1, where customizations mainly concerned the introduction of a hierarchical structuring of the goals and a simplified version of some intermediate working documents. In particular a unique report (the *GQM plan*) has been delivered. It is organized in such a way to include both methodological and operative issues, but, at the same time, keeping them properly distinguishable and upgradeable. In fact, the *GQM plan* resulted to be a very "dynamic" document, in the sense that several iterations were necessary to best take into account the new aspects that came to appearance during the overall research activity. At the end, as expected, the adopted approach, although originated from the waterfall-like sequence of steps of Fig. 1, proved to be a cyclic set of iterative steps, eventually leading to a remarkable and significant result.

6 References

[1] Victor R. Basili "Software modelling and measurement: The Goal/Question/Metric paradigm. Technical Report CS-TR-2956, Departement of Computer Science of Maryland September 1992.

[2] Victor R. Basili, "Models and Metrics for Software Management and Engineering," IEEE Computer Society Press, IEEE Catalog □o. E□O-167-7, (343 pp), 1980.

[3] Ip, A., Morrison, I. "Learning Objects in different pedagogical paradigms" (Online), http://users.tpg.com.au/adslfrcf/lo

[4] SCORM http://www.adlnet.org

[5] IEEE ISO/FCD 9126-1.2 Information Technology – Software Product Quality - Part 1:Quality model, 1998

[6] IEEE WD 9126-3 Information Technology – Software Quality Characteristics and Metrics – Part 3 : Internal Metrics, 1997.

[7] IEEE ISO/IEC WD 9126-4, Information Technology - Software Product Quality - Part 4 Quality in use metrics, 1999

[8] B. Fadini, P. Maresca, S. Russo, “Un approccio alla valutazione di piattaforme per l’e-learning basate su standard di qualità ISO”, in *E-Learning metodi, strumenti ed esperienze a confronto*, Didamatica 2002 - Informatica per la Didattica – a cura di A. Andronico, A. Chianese, B. Fadini, pp. 107-132, □apoli, 14-15 Febbraio 2002, Liguori Editore (in Italian)

[9] A. Fuggetta, L. Lavazza, S. Morasca, S. Cinti, G. Oldano e E. Orazi “Applying GQM in an industrial software factory. ACM Transactions on Software Engineering and Methodology, Ottobre 1998.

[10] B. Fadini, P. Maresca, P. Prinetto, C. Sanghez, G. Santiano , “GQM plan for e-learning platform” CI□I Internal Report, 2005.

Models of pragmatics of man-machine interaction (Perspectives and problems in the Elementary Pragmatic Model experimentation)

Sabina Bordiga, Luigi Colazzo, Francesco Magagnino, Daniela Malinverni, Andrea Molinari

Università degli Studi di Trento Dipartimento di Informatica e Studi Aziendali

Via Inama, 5 38100 Trento Italy - Tel +39 0461 882144 Fax +39 0461 882124, Emails: bordsab_2@yahoo.it, colazzo@cs.unitn.it, francesco.magagnino@ewave.it, dmalinve@economia.unitn.i , amolinar@economia.unitn.it

Abstract

Learning processes are not homogeneous and valid for all the students. Different individuals learn in different ways under the same cultural conditions. Moreover, learning processes are conditioned by age and circumstances. The teaching personalization is not practically possible in educational systems not supported by computers, because of economic and organizing matters. However, it's possible to personalize educational actions according to the features of single students throughout adaptive software systems. e_learning systems are one of the more promising application fields among the techniques that allow a software system to adapt to the preferences and needs of its users. In fact an effective e_Learning system should threat differently every single user. For this and other reasons, an e_Learning system should be built to model itself on the individual features of the user, recommending and suggesting educational paths that are suitable for the learning subject

Introduction

This work presents the state of our research in the attempt to use models coming from pragmatics of man-machine interaction. The final aim of this study is to prove that it's possible to build new adaptive software systems that extend users behaviours interpretation by including a description of the pragmatics of their interaction.

We are interested in the e-learning applicative field. In fact, in these systems it's a desirable feature the system adaptability to the different characteristics of learning subjects, to the different leaning styles and, finally, and to the process context to be a desirable feature.

The majority of e-learning systems are built as Web applications. In these cases it's practically impossible to know the final user needs. In fact, it's well known that the final user can be very different each others in training environment, current language, culture etc. Thus, it's not technically possible to get the formal description of their features, in order to lead the process of educational software production. Perhaps it could not even be useful to have this description because it's hard to hypothesize for any of us the way we would interact with something unknown (what we want to learn, as a matter of fact, it's unknown). So, the design requirements of these systems are given by the teacher, who knows the subject of his course and s/he is able to define how to teach it: so s/he able to tell to the system designer the project requirements. But the teacher perspective could be very different form the leaner perspective.

Many authors [Barbe & Milone, 1981], [Corno & Snow, 1986], [Felder, 1988], [Pask, 1988] observed that learning processes are influenced by cognitive constants typical of any student. In [Felder, 1988] the hypothesis is that there are personal learning/teaching styles which can be traced in a four dimensional space, defined by opposite couples (active/reflective, sensitive/intuitive, visual/verbal, sequential/global).

The same authors [Felder, 1993] underline that also teaching styles are different and this strengthen what we had just said about the difficult to define formal requirements for the e-learning applications building.

Thus, the hypothesis to build adaptable e-learning systems seems very promising. The hope is that systems like that could give us the possibility to personalise teaching techniques on the single student needs (currently, adaptable systems use stereotypes to interact with students).

For this purpose we are studying the opportunities given by EPM [Lefons et al., 1977], [Silvestri et al., 1980]. The work presents the present state of this research highlighting the problems we met during the experimentation and the perspectives that this approach could open in the building of new generation adaptive systems.

In the paper we shortly discuss why we mean to extend the traditional interpretation models of user-system interactions (at the basis of any adaptive system) by including a pragmatic interpretation of the interaction. In fact present models are based only on semantics and syntax, two of the three aspects that Charles Morris used to simplify the communicative action [Morris, 1938] .

Then we discuss the desirable features that a model of pragmatics of man-machine interaction should have. In fact an extension in this direction needs, to be effective, formal and easy to calculate models.

Finally we present our experience trying to use EPM as a possible interpretative model. In this part of the paper we present a synthesis of EPM features, highlighting how this model seems to have many features desirable for our aims. After this, we show examples of the way this model can be used to interpret simple interaction processes. Moreover, we talk about the problems faced when a model like EPM is transferred from the original context (human interaction) to a new communication context (human computer interaction).

At the end, we describe a preliminary experiment that has the aim to verify the applicability of the model to situations of real use of an e-learning system.

1. User models extension to the pragmatic aspect of communication

It's empirically possible to note that computer users show peculiar behavioral modules while using software applications: these modules seem behavioural constants. For example, if a user taps combinations of keys to execute some actions, she will tend to use them also when she changes software environment, and she'll be disturbed by a software which modifies the syntax of keys combinations. The nature of these behavioural constants seems to be independent of the syntax of the interaction. The user tries to adapt the system to her own habits, independently of the type of syntax proposed by the software to express them. At the same time, these behavioural constants seem to be independent of the semantic of the task, for it's easy to note that users show these constants in very different situations

Most in general, in the studies about interpersonal relationships (psychiatric and psychological studies), it is well known that communication is characterized, besides syntactic and semantic aspects, also by the influence that participants experience each other. This last aspect, i.e., the effect that a communicative interaction provokes in interagents, is definable as the pragmatic aspect [Watzlawick et al., 1971].

Following this tri-partition of communication (considering that syntax, semantics and pragmatic are clearly separated in theory, but strictly interlaced during concrete interactions), changing syntactic and semantic components, the non-changing component could be the pragmatic one. The studies about the pragmatic of relationships according the theory developed by the Palo Alto Instituted (inspired by the relational approach to the mind by Bateson) laid stress on the fact that every subject has her own way to interpret the syntax and the semantics of relationship, and this interpretation is based on the pragmatic of the relationship that she keeps up with the world or even with a single subject. At the same time, the main pragmatic interpretation defines a relational style of the subject, resulting from the history of the interaction processes with reference people [Watzlawick et al., 1971]. The subject tends to reproduce constantly her interpretative scheme of relationships, also when the syntax for relationship or the semantics change. We assume that the behavioural constants empirically observed in users of software applications deal also with these pragmatic interpretations of the relationship. If that was true, then it could be possible to identify these pragmatic redundancies and use them to develop adaptive mechanisms in software systems.

The possibility to use the descriptive constants of users behaviour in order to characterize them and offer a personalized service is the premise of the interpretative studies on adaptivity of web applications. There is a wide literature about the possibility to provide the software systems with adaptive mechanisms which allow the systems themselves to personalize their behaviour according to the users needs [Foster, 2000]. An initial classification of these systems is made by the distinction between user-adaptive systems properly said (the adaptive mechanisms are completely automatic) and user-adaptable systems (the adaptive mechanisms are derived by user choices) [Teltzrow & Kobsa, 2004]. In this first type, the system must have a data structure for every user who interacts with the system and in this structure the history of the interaction must be condensed in a shape (behavioural model) useful to the system to implement its strategies. This data structures are known in the literature as User Models (UM) [Kobsa, 1990]. A UM is a data structure which gathers information on the interactions of a user with the system. Typically, the UMs are obtained on the basis of the knowledge about: interests and aims of the user [Kulls, 2000], competences about the use or the contents of the system, history of the interaction of the user with the system. In [Teltzrow & Kobsa, 2004] is reported a list of about sixteen classes of data sources about the user, generally used in the building of UM. The unprocessed datum itself can not be used immediately in the adaptive processes of a system. In order to change those data in an input for the processes of adaptation of the system, it's necessary their interpretation. It is to say that the variability is abstracted through formal processes of generalizations known as stereotypes [Rich, 1979]. Formally a stereotype is a structure of generalization which describes different characteristics of an hypothetic user. Any characteristic is associated to a trigger. The unprocessed data of a user, abstracted through this interpretation, build the UM of the user for the system.

The literature indicates three basic steps to prepare the development of an adaptive system: the first regards the methodology used to obtain the UM; the second step, consequently to the chosen methodology, is about the type of data which have to be classified and stored [Brusilowsky et al., 1996]. Finally, the third step concerns the personalized presentation for the information, which is depending on the goal of the system. The personalization occurs with different paths in texts, images, lexicon and in the User Interface in general. It can occur also following rules of ubiquitary adaptivity for the presentation on different platforms and hardware systems. In this work, which has the purpose of presenting a current experimentation, we are going to talk only about the first step of this process.

2. A relational approach to the building of UM

According the approach mentioned in the introduction, while describing the relationship between two interacting subjects (A and B) it is better to consider that this relationship is the dynamic development of two distinct interpretations: the interpretation given by subject A to her relationship with B, and the interpretation given by the subject B to her relationship with A. We could also say that there is always a sort of "inverse vision" of the relation. In the case of two human beings, descriptions must be inferred by the observation of the behaviours of the two subjects.

Instead, in the case of the relation between a user and a software system, the situation is simplest. In fact, the interpretation of this relation is fully specified by its finite states automaton, where there is a description of all the possible states of the system and the state transaction that a system can or has to activate answering to a user inquiry. Its formal description is the Pragmatic System Model (PSM). Instead, the description of the interpretation given by the user to the system must be obtained by the observation of the user behaviours. The formal description of this interpretation is a Pragmatic User Model (PUM).

In order to build the PSM of the system and the PUM of the subject we need a theoretical model to interpret the behaviours of the two actors of the relation. A model like that must have some desirable characteristics:

A. The model must be able to describe both the PSM of a specific software system and the PUM of a human subject by a standardized formulation.

B. The model must allow to obtain the formal description of the PUM by the real behaviour of the user with the system.

C. The model must be predictive, i.e., must be able to foresee the effects provoked by a change of the behaviour of one of the two actors of the relation on the other one.

D. The model must allow the system to adapt itself to the relation shown by the user, i.e., it must suggest to the system a behavioural change right to an effect pragmatically positive on the relation.

E. The model must not have any kind of hidden interpretative prejudice on the rational behaviour of the user.

The tri-partition of a communicative act in syntax, semantics and pragmatic is due to Charles Morris [Morris, 1938]. Two partially different interpretations derived from Morris' definition of the term pragmatic and, in a way, they are both easy to be found in computer science literature. EPM is about one of the two, the one most used in studies of psychopathology started from Gregory Bateson studies and then from his followers of the Palo Alto Institute.

This definition of the term "pragmatic" is different from the one commonly used in linguistic [Levinson 1983]. Again, both these definitions are different from the one easy to meet in the literature about UM. In the current literature about UM [Strachan et al., 1997], [Fernandez-Manjon et al., 1998] and in the studies about Pragmatic Web [Dumbill, 1999], the adjective "pragmatic" is synonymous of ergonomic, both as for a pragmatic user model and a pragmatic approach to the building of adaptive sites. "Pragmatic" substantially means something very functional to the goal. Then, according these studies, a pragmatic user model is a model which allows to have information on the user behaviour, or a possible interpretation useful to the knowledge of the user-system interaction. Instead, what we mean by the term pragmatic in this work is to be found in the aspect of action that a behaviour of a person has as part of a system.

3. Elementary Pragmatic Model (EPM)

In this section we are going to present only a short synthesis of EPM, inviting the readers to refer to the bibliography of the model easy to find on line in the home page of one of the authors of the model [De Giacomo, 2003]. The model assumes that interacting subjects are definable as a universe of propositions (the world of the subject) and a mechanism of change. The effect of a single communicative act between two subjects leads to a change in the state of the world of A and in the world of B. Following the basic idea of the pragmatic theory by which every behaviours is a communicative behaviour (i.e., it is not possible not to communicate), the elementary pragmatic model simplify the communicative interaction between two subjects in a triple of propositions:

1. proposal of one of the subjects
2. reply of the other subject
3. final result of the exchange.

Semantics, the content of the interaction, is not enough: if a person asks to go out and have a pizza or to set a sentence down in a board of directors, her communicative action has the pragmatical meaning of a proposal. At the same time, the reply (people involved answer that they too want to have a pizza or the board of directors doesn't want to set the sentence down) and the result of the exchange (they don't go out for a pizza or they don't set the sentence down) are independent of semantics.

EPM isolates a temporal pause and attributes a binary value to any proposition, considering it the only and most elementary way to describe a communicative exchange. For example, in the case of a board of directors, the model gives the value (1) to the request to set the sentence down, it gives a different value (0) to the reply of the board not to do it and it gives a value different from the value of the proposal (0) to the result of the interaction (they don't set the sentence down). In fact, according to the model, the state of the subject is determined by its world or by a limited but indeterminate set of propositions and by a mechanism of change of the observed subject. The hypothesis of the model is that this mechanism can be obtained by the observation of her interactions. In order to obtain a quantitative description of the mechanism, the model observes a sequence of elementary interactions, gets the observed frequency for every possible triple and uses it as a valuation of the occurrence probability of four coordinates of the interaction according the following calculus scheme

U1= n (0, 0, 1)/n (0, 0, 0) + n (0, 0, 1)

U2= n (0, 1, 1)/n (0, 1, 0) + n (0, 1, 1)

U3= n (1, 0, 1)/n (1, 0, 0) + n (1, 0, 1)

U4= n (1, 1, 1)/ n (1, 1, 0) + n (1, 1, 1)

Where n is the number of observed sequence for every triple. The model gives a pragmatic interpretation to these coordinates:

P(<0, 0, 1>) (Anti-function). In an interactive exchange the action project of one of the subject (0) obtains a positive reply (0) form the other actor, but the result of the interaction (1) is something different from what has been exchanged during the interaction.

P(<0, 1, 1>) (Agreement). The action project of one of the subjects (0) obtains a new proposal (1), and the result of the exchange is the agreement of both subjects on the new proposal (1).

P(<1, 0, 1>) (Maintenance). The action project of one of the subject (1) receives a negative answer (0) by the other subject, but the result of the exchange is the maintenance of the first proposal (1).

P(<1, 1, 1>) (Cooperation). The proposal of one of the subjects (1) has been accepted by the other participant (1) and it becomes the result of the exchange (1).

The four coordinates are independent each other while during interpersonal relationships are always present. For this reason the pragmatic redundancies of a subject can not always be estimated only by the coordinates. However, these last ones can be used to obtain a relational style of the subject: the probability of the subject to use one of the sixteen Boolean functions at two values. Using the principle of the maximum information, these functions can be calculated using the calculus formula of table 1.

F	Fn Value	Fn meaning
f_0	$(1-u_1)(1-u_2)(1-u_3)(1-u_4)$	No, no, no, no, no...
f_1	$(1-u_1)(1-u_2)(1-u_3)\, u_4$	Our relation is based only on what we share
f_2	$(1-u_1)(1-u_2)\, u_3\,(1-u_4)$	I only accept what is exclusively mine
f_3	$(1-u_1)(1-u_2)\, u_3\, u_4$	I maintain my worldview
f_4	$(1-u_1)\, u_2\,(1-u_3)(1-u_4)$	Only what is exclusively yours interests me
f_5	$(1-u_1)\, u_2\,(1-u_3)\, u_4$	I enter your world with what we have in common
f_6	$(1-u_1)\, u_2\, u_3\,(1-u_4)$	Everything in my world and in yours is part of our relationship as long as is not shared
f_7	$(1-u_1)\, u_2\, u_3\, u_4$	Our relationship is based on the union of our worlds
f_8	$u_1\,(1-u_2)(1-u_3)(1-u_4)$	I only have relationships with what is alien to me and to you
f_9	$u_1\,(1-u_2)(1-u_3)\, u_4$	What we have in common and what we have is alien
f_{10}	$u_1\,(1-u_2)\, u_3\,(1-u_4)$	I am an argumentative person
f_{11}	$u_1\,(1-u_2) u_3\, u_4$	I'm interested in everything, also outside our relation, as long as it is not specifically yours
f_{12}	$u_1\, u_2\,(1-u_3)\,(1-u_4)$	What is important for me is your world exclusively, and external elements
f_{13}	$u_1\, u_2\,(1-u_3)\, u_4$	I enter your world using external elements
f_{14}	$u_1\, u_2\, u_3\,(1-u_4)$	Everything, also what is alien interests me, except for what we share
f_{15}	$u_1\, u_2\, u_3\, u_4$	Yes, yes, yes, yes...

Table 1 Calculus mechanism of the observed use frequency of the 16 functions, beginning by the observed values of coordinates

The knowledge of an interaction style of a subject could supply useful information on how to interact with her. The knowledge of the relational pattern of a subject allows to predict the best function to interact with that subject. The way a function changes in interaction with another function, defining the consequent change of the interaction mechanism, is represented by the paradoxes table contained in the model.

The EPM satisfies all the five characteristics useful to interpret the man-machine interaction:

The model does not presuppose that interacting subjects are human being, so it can be used to describe behaviours of an interactive program. The descriptions of the PUM and the PSM can be formally similar (pattern of sixteen functions): the first description is obtained by the observation of the user behaviour according the interpretation given by the model, the second one by the fact that we exactly know the behaviour of the system. That satisfies the A characteristic.

The model allows to interpret the relational style of the user during her interactions with the system: known the pragmatic system model (PSM), the actions of the user inside the system can be codified, gathered and elaborated according the EPM to the reaching of the pragmatic model of the user herself (PUM). This aspect satisfies the B characteristic.

The Elementary Pragmatic Model is a predictive model: thanks to the paradox table, it's possible to foresee how an interactive style changes when it meets another one. (C Characteristic)

Knowing in advance the style resulting from the meeting of the two subjects' styles would allow the system to adapt to the style shown by the user: on the basis of the user pattern the system can adapt to that style according a positive resulting. That should make possible to satisfy the D characteristic.

The Elementary Pragmatic Model (EPM) was defined at the end of 70s by a group of physicist, psychiatrics, mathematicians and informatics on the basis of the relational theories of the Palo Alto Institute, as tool to interpret psychopathologic behaviours. For this reason the model doesn't have any assumption of the rational behaviour of subjects. That satisfies the last characteristic of the desirable characteristics of section 1.

The application of such model to electronic communication was proposed in some works applying the model to the interpretation of conversations through electronic mail systems [Colazzo et al., 1991]. In recent past the idea of the use of the model to interpret the relationship between users and Web sites has been proposed in [Colazzo et al., 1999]. We now re-propose the concrete experimentation of this approach.

4. EPM application to a software system

In the previous section we described the meaning that EPM authors gave to the four coordinates of the model. In this section we face the problem to the application of the four coordinates to an interactive software system.

The first coordinate is an antifunction, it is to say that after a user proposal the system replies accepting the proposal but the result of the action is something different from the proposal itself. In a perfect software system this situation should not happen. Real systems are not always perfect, then there are the probabilities that an antifunction can happen. According to a state diagram perspective, the situation can be described in two ways.

In the first case, the user ask explicitly something: the system answer by the acceptance of the proposal, but it produces a different result.

In the second case, the user does not propose anything and the system seems to accept the not action of the user, but after few moments the system do something by itself.

In the Fig1 diagram, it is shown an explicative fragment of the first antifunctional situation. The user opens the Menu File and the system visualizes it correctly (according the model codification: 1). The user chooses the option Menu Open File and the system visualizes the selection mask of the files (according to the model codification: 1). The user chooses to open the X file, but the system opens the file y (according to the model codification: 0).

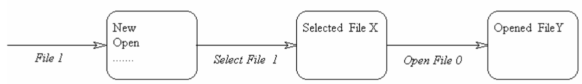


Fig1 Example of an antifunctional behaviour of the system, deriving from a system error

In the Fig2 diagram, it is shown an exemplificative fragment of the second antifunctional situation. The user does not choose any of the options that the current state of the system foresees (codified by the model by 0). In these conditions, the system does not act (codified by 0). Then after the session time slice, the system changes state and set a login state (codified by 1). This behaviour is typically implemented for safety reasons. In this it is avoided that an user can leave a work session opened while s/he is far from his pc.

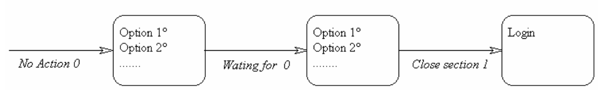


Fig2 Example of an antifunctional behaviour deriving by system choices.

Anyway, there is a certain number of situations where the system does not behave as the user believe. Although the user perceives these cases as antifunctional cases, they are not antifunctional according to the system perspective. In fact, what happens is that the actions done by an user in order to achieve an objective are not the right actions according to the system logic. It is to say that the user tries to do something but s/he does it in the wrong way. If many users fall in the same pragmatic situation, the tracing of these cases underlines the need to revise the system interface.

The second coordinate is the agreement coordinate and it happens when the system provides an advice or a refusal to a user request, and the user accepts or require again according to what the systems wants. Also in this case the system could react to an explicit user action or to an user non action. It follows that two different situations of agreement can happen. Under the state diagram perspective, the situation can be described in two ways.

In the first case, the user asks something to the system. The system reacts refusing or advising an alternative solution. The user follows the system request or accepts the suggestion.

In the second case, the system reacts to a non action of the user, trying to obtain the user attention. The user reacts and the systems leaves the non action state.

In the diagram of Fig3 there is the example fragment of the first situation. In this case the user asks to fill a data entry form (codified by the model by 1). While filling the form, s/he forgets to insert determinant data (example: the e-mail address in the registration to a Web application service). The system refuses the input and gives an error message (codified by the model by 1) and takes the user back to the form filling state (codified by 0)

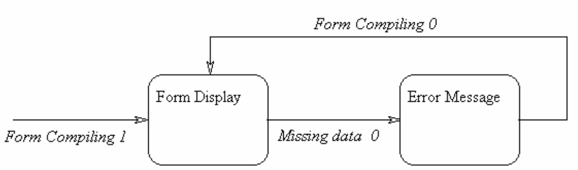


Fig3 Example of an agreement behaviour of error correction

In the Fig4 diagram, there is an agreement situation generated by a non action of the user. The situation is not like the situation of Fig2. The user is not using the system (codified by the model by 0) but in this case the system produces a message at the end of the time slice session and press the user (codified by the model by 1) for a decision. The user can decide to keep the session open (codified by 0) or close the session and the system goes back to the login state (codified by the model by 1).

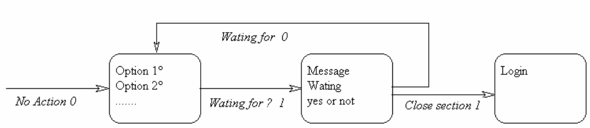


Fig4 Example of the behaviour pressed by the system asking a decision to the user.

EPM defines the third coordinate as maintenance. In the interaction with a software system this situation happens where the system tries to warn the user about the consequences of his action or non action (see Fig4 if the user decides to keep the working session open at the end of the time slice as an example of non action maintenance). Normally these interaction situations happen when the user tries to do a possible action with irreparable consequences. Example: the cancellation of not read e-mail messages (typical case of spam messages) or the upload of potentially damaging files.

An example of limited states automaton is the one in Fig5, where the system reacts to the cancellation request of not read e-mail messages (codified by the model by 1) by asking the confirmation of the action (codified by the model by 0), and the user replies keeping his former action project (codified by the model by 1).

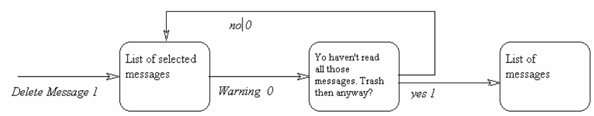


Fig5 Example of user maintenance behaviour

The fourth coordinate anticipated by EPM is the Cooperation Coordinate. In the man-machine interaction this coordinate is predominant or it's the coordinate that grants the system working. It's not hard to find examples of this coordinate in a well working system behaviour. Two kinds of cooperation are possible. In the first case the user asks to the system to act and the systems does it. In the second case, the user doesn't ask anything to the system and the system cooperates doing anything.

In the two possible types of cooperation, the first one is the more frequent while the second one is almost totally absent. The reason for this is that the software systems are objects built to do tasks and then there are few reasons to implement complete mechanisms of non action.

In the Fig6 diagram it's shown a typical case of Cooperation. The example is similar to the one in Fig1, but this time no error happens and the opened file is the required file.

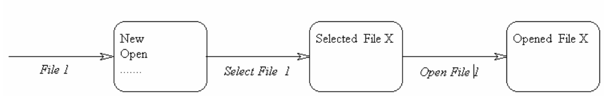


Fig. 6 An example of cooperation behaviour with the user.

5. Experiments with EPM (first results)

In order to verify the possibility to use EPM for a web application our research groups starter a preliminary experimentation. The results of these experiments are encouraging so far, but not definitive. Our objectives in this first phase are to find an answer to the following questions:

- *Are PUMs obtained by the model application different each other?*
- *Does PSM give us a credible description of the pragmatic behaviour of the system?*
- *Does the PUM of a specific user correspond to the pattern that other psychological instruments give?*
- *Do PUM & PSM describe a real situation of user-system interaction?*

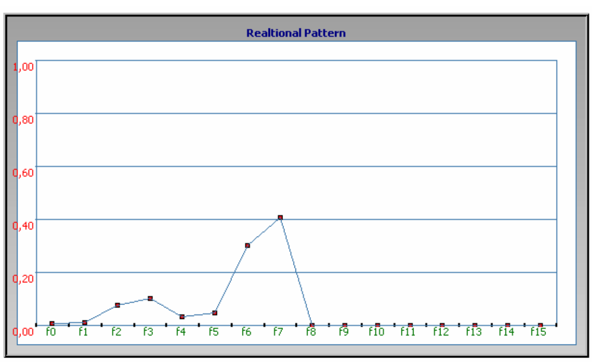
For the first experimentations we built a small software system , an application for the realisation and managing of a forum. We obviously know the limited states automaton of the application. It is to say that we know all the possible user-system interactions. In fact these interactions are easy to be traced by the state transactions of the diagram. The state transactions are codified according to EPM and the interactions of the participant subjects are recorded as a binary vector.

The result of this preliminary experiment underlined a certain number of issues that partially gave us an answer to our preliminary questions and partially made us understand some limits of our initial suppositions.

First, we realized that we need to codify the non action as we shown in section 4. At the beginning, the non action were not actually codified, so in many situations the resulting pattern gave important errors.

The first of the questions we made comes from the fact that a software system could have a predominant effect on the possible user interactions. It is to say that we wandered what happens if the system will compel the users to show the same relational style: the style imposed by the system.

As we hoped, the patterns of the users resulted different each others (Fig7). Nevertheless, patterns show important anomalies as shown by Fig7. In fact the two pattern represent a) the behaviour of a normal user in the system and b) the web master behaviour while committed in a debugging task. In the first case there are only functions form 0 to 7, while in the second case only functions from 7 to 15.



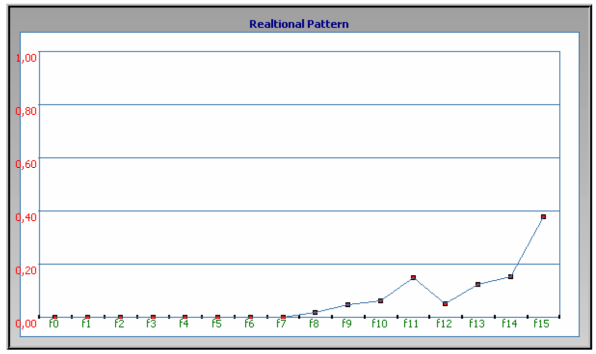


Fig7 Two different patterns

The second question has a partial answer too. The PSM of the system seems to be stable: it's not changed during the increasing number of user interactions. Nevertheless, we used a reactive software, not an adaptable one. This leaded to a prevalence of the Function number 15, the function of a system that just says yes to everything asked by the user. This is not surprising: a system is built with an aim and if the software is a good software should do what we tell it to do (see Fig8).

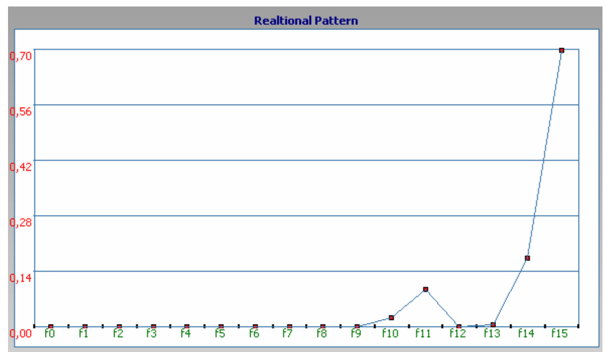


Fig8 The relational pattern of the system

Then, to have a definitive answer to our question it should be necessary to experiment EPM on a traditional adaptable system, based on the semantic and syntactic interpretation of the users behaviour. This requires the building of another test-bed, and this is what we are doing in this period.

Finally, the two final questions are not yet a definitive answer. Two different reactive psychological tests are connected to EPM, produced independently by two different research groups. The first is known as SISCI II , the second one as Kaltest [Pereira & Ferreira 1983]. These two reactive tests are conceptually similar but different in the approach. In synthesis, SISCI II works as follow: a certain number of abstract imagines [Holtzmann 1958 1961 1972] are shown to the subject and s/he has to choose some and refuse the not chosen ones. The choose of an image is codified by 1, the refuse by 0. After that, a random selection of images, declaring that they have been chosen by the majority of the subjects, are presented to the subject. Finally, the subject has to choose again the images, showing the first set of images but in a different order.

The opinion change or its maintenance is recorded allowing to deduce the subject pattern. The images set used in the SISCI II is pre defined, while in Kaltest the images are automatically generated by a program and are different in different experiments.

To give an answer to the final questions it's necessary to use one of these reactive tests. At the moment, our choice tends to Kaltest because it allows us to pre calculate the user pattern and verify it with the pattern resulting from the system. Currently, we built a Kaltest version and we are setting it on our needs.

Conclusions

In this work we presented the state of our experiments that tend to verify the opportunity to adopt pragmatic models in the building of user model in e-learning systems. We discussed about the usefulness of this extension. We shortly described the theoretical model of the interaction developed in the psychiatric field. Finally we discussed the applicability of the model to the man-machine interaction and we described the results of our preliminary experimentations.

References

[Barbe & Milone, 1981] Barbe, W.B. Milone M.N. (1981) What We Know About Modality Strengths. Educational Leadership, Feb. pp. 378-380.

[Brusilowsky, 1996] Brusilowsky P. (1996) Adaptive Hypermedia: an Attempt to Analyze and Generalize. In P. Brusilowsky, P. Kommers & N. Streitz (Eds.), Multimedia, Hypermedia and Virtual Reality, Berlin:Springer-Verlag, pp 288-304.

[Corno & Snow, 1986] Corno, L. Snow R.E.(1986) Adapting Teaching to Individual Differences Among Learners. In M. Wittrock, Ed., Handbook of Research on Teaching. Macmillan, New York.

[De Giacomo, 2003] De Giacomo P. (2003) Elementary Pragmatic Model References Available on Web: http://www.pierodegiacomo.it, last access June 08 2004.

[Dumbill E., 1999] Dumbill E., 1999 The pragmatic Web. Available on Web: http://usefulinc.com/articles/pragmatic.html, last access June 02 2004.

[Felder, 1993] Felder, R. (1993) Reaching the Second Tier: Learning and Teaching Styles in College Science Education. In J. College Science Teaching, 23(5), 286-290.

[Felder, 1988] Felder, R. (1988) How Students Learn: Adapting Teaching Styles to Learning Styles. In Proc., Frontiers in Education Conference, ASEE/IEEE, Santa Barbara, CA, 1988, p. 489.

[Fernandez-Manjon et al., 1998] Fernandez-Manjon B., Fernandez-Valmayor A. e Fernandez-Chamizo C., (1998), Pragmatic User Model Implementation in an Intelligent Help System, available on Web: http://www.blackwell-synergy.com/links/doi/10.1111/1467-8535.00053/abs/, last access Giugno 02 2004.

[Foster, 2000] Foster, C. (2000) The Personalization Chain. In Jupiter Communications, Site Operations, Vol. 3, 2000.

[Holtzman W.H., Thorpe J.S., Swarz J.D:, Herron E.W.,] Inkblot perception and personality: Holtzman Inkblot technique, Hogg Fundation for Mental Health, University of Texas Press, Austin, 1961

[Holtzman W.H.] Il metodo psicodianiostico di Holtzman, trad. Silvano Chiari, Firenze, Edizione Organizzazioni Sepciali, 1972

[Holtzman W.H.] the Holtzman inkblot technique (from A and B), New York, Psychological Corporation, 1958

[Kobsa, 1990] Kobsa A., (1990) User Modeling in Dialog Systems: Potentials and Hazards. In AI & Society 4(3), 214-240.

[Kulls, 2000] Kules B.(2000) User Modeling for Adaptive and Adaptable Spftware Systems. Available on Web: http://www.otal.umd.edu/UUGuide/wmk/, last access June 02 2004.

[Lefons et al., 1977] Lefons E., Pazienza M.T., Silvestri A., Tangorra F., Corfiati L., De Giacomo P. (1997) An Algebraic Model for Systems of Psychically Interacting Subjects. In Proc. of I.F.A.C. Workshop, Compiègne, France, in "Information and Systems", Oxford, Pergamon Press.

[Levinson, 1983] Levinson S. C. (1983) Pragmatics, Cambridge University Press.

[Morris, 1938] Morris C. W. (1938) Foundations of the theory of signs. In International Encyclopaedia of Unified Science, Chicago, University of Chicago Press. 77-138.

[Pask, 1988] Pask, G. (1988) Learning Strategies, Teaching Strategies, and Conceptual or Learning Style. In Schmeck, R., Ed., Learning Strategies and Learning Styles, Plenum Press, New York.

[Pereira & Ferreira 1985] (1985) Applicazioni del modello di De Giacomo e Silvestri in psicologia in Silvestri A., Cipolli C.

[Rich, 1979] Rich E. (1979) User modeling via stereotypes. Cognitive Science, 1979, 3: 329-354.

[Silvestri et al., 1980] Silvestri A., De Giacomo P., Pierri G., Lefons E., Pazienza M.T., Tangorra F., (1980) A Basic Model of Interacting Subjects. Cybernetics and Systems: An International Journal, 11, 113-127.

[Strachan et al., 1997] Strachan L., Anderson J., Sneesby M. and Evans M., (1997) Pragmatic User Modelling in a Commercial Software System. Proceedings of the Sixth International Conference on User Modelling, Sardinia, Italy, pp. 189-200

[Teltzrow & Kobsa, 2004] Teltzrow M., Kobsa A. (2004) Impacts of user privacy preferences on personalized systems. In: C.-M. Karat, J. Blom and J. Karat, eds: Designing Personalized User Experiences for eCommerce. Dordrecht, Netherlands: Kluwer Academic Publishers.

[Watzlawick et al., 1971] Watzlawick P., Beavin J. H., Jackson D. D., (1971) Pragmatica della comunicazione umana. Astrolabio, Roma.

Ontologies for E-Learning

F. Colace and M. De Santo

DIIIE – Università di Salerno, Via Ponte don Melillo, 1 I-84084, Fisciano (SA), Italy
{colace, desanto}@unisa.it

Abstract

Distance education represents today an effective way for supporting and sometimes substituting the traditional formative processes, thanks to the technological improvements achieved in the field in recent years. In the open scientific literature, it is widely recognized that an important factor of this success is related with the capability of customizing the learning process for the specific needs of a given learner. This feature is still far to have been reached and there is a real interest for investigating new approaches and tools to adapt the formative process on specific individual needs. In this framework, the introduction of Ontology formalism can improve the quality of formative process, allowing the introduction of new and effective services. The use of Ontologies can lead to important improvements in the definition of courses knowledge domain, in the generation of adapted learning path and in the assessment phase. In this paper, we discuss such improvements related to the introduction of Ontologies formalism in the E-Learning field and we present a novel algorithm for Ontology building through the use of Bayesian Networks. Finally, we show its application in the assessment process and experimental results obtained.

1. Introduction

The development of adaptable and intelligent educational systems is widely considered one of the great challenges in scientific research. Among key elements for building advanced training systems, an important role is played by methodologies chosen for knowledge representation. For example, the introduction of standardized tools for defining a set of well-structured concepts can highly improve interoperability and information sharing between complex systems. In literature, a set of concepts and their relationships is commonly called ontology [1]. Ontology is one of the most effective tools for formalizing the knowledge shared by groups of people. In the E-Learning realm, ontology can easily manage the knowledge domain of a course and allow a more detailed organization and adaptation of the learning path of students [2][3]. Nevertheless, ontology building process is neither trivial nor easy [4]. By way of illustration, let us discuss the common case of the use of ontologies for representing the subjects of a course and the relationships among them. When asked to describe their course, teachers regularly provide very intricate representations, so having to manage ontologies that are neither easy to interpret nor to use. Very often, relations among the concepts and their semantic values look more similar to complex puzzles than to useful working tools. It is easy to imagine the consequent difficulty in checking the validity of such descriptions . One way to face the quoted problem is to make available to the user automated tools for building and validating ontologies [5][6]. In this paper, we propose a method for ontology building that can be applied to knowledge domain related to University Curricula. In such a context, we have a powerful source of evidence: the evaluation tests performed at the end of the course. Usually, teachers design evaluation tests taking into careful account the sequencing and preparatory links of course subjects. So, in our opinion, those tests and the answers given to them by students can be used to individuate the ontology of the course. To this aim, we show how to use Bayesian Networks for easily mapping ontologies and present a novel algorithm for building "lightweight" ontologies through them. Furthermore, we illustrate the application of this method in the assessment phase. Namely, we realized a tool that builds the best assessment strategy according to the information inferred by the analysis of questionnaires. The paper is organized as follows: in Section 2 and 3, we give some details on ontologies and their mapping through Bayesian Networks. In section 4 we describe the proposed approach for the ontologies building process. In section 5 we provide the motivations and the details of assessment tool based on ontology and Bayesian Networks. Finally, in the last section we draw conclusions and indicate future directions of our research.

2. Ontology

The concept of ontology is originally taken from philosophy where it means a systematic explanation of being. In recent years, however, this concept has been introduced and used in different contexts, thereby playing a predominant role in knowledge engineering and in artificial intelligence [7]. In 1991, Neches stated that ontology defines the basic terms and relations comprising the vocabulary of a topic area, as well as the rules for combining terms and relations to define extensions to the vocabulary [1]. Later on, Gruber, in the context of knowledge sharing, used the term to refer to an explicit specification of a conceptualization [8]. In the field of computer science, ontology represents a tool useful to the learning processes that are typical of artificial intelligence. In fact, the use of ontologies is rapidly growing thanks to the significant functions they are carrying out in information systems, semantic web and knowledge-based systems. The current attention to ontologies paid by the AI community also arises from its recent interest in content theories, an interest that is greater than the one in mechanism theories. In this regard, Chandrasekaran [7] makes a clear distinction between these theories by asserting that, though mechanisms are important since they are proposed as the secret of making intelligent machines, they can not do much without a good content theory of the domain on which they have to work. Besides, once a good content theory is available, many different mechanisms can be used to implement effective systems, all using essentially the same content. Following this point of view, ontologies are content theories, since their principal contribution consists in identifying specific classes of objects and relations existing in some knowledge domains [9]. Ontological analysis, therefore, clarifies knowledge structures: given a domain, its ontology represents the heart of any knowledge representation system for that domain. Another reason for creating and developing ontology is the possibility of sharing and reusing knowledge domain among people or software agents. In general, ontology is a complex structure made up of a series of elements, each of which is composed of a kind of Relation and a series of related Concepts. For example, as far as concerning University Courses, by means of an ontology built by the teacher, it will be possible to describe the knowledge domain, the subjects constituting it, the relations among the various subjects, as well as methodologies and means with which they are presented. These explicit specifications help users to understand what specific terms signify in a given domain [2] and reduce terminological and conceptual ambiguity. Ontologies can be represented as taxonomic trees of conceptualizations: they are general and domain-independent at a superior level, but become more and more specific when one goes down the hierarchy. In other words, when we move from the highest taxonomic levels to the lowest ones, characteristics and aspects typical of the domain under examination are showed. In order to point out this difference in literature they are called heavyweight (deeper) and lightweight (advances) ontology respectively [3]. In this paper, we will adopt the last one approach keeping in mind this definition of ontology: "ontology may take a variety of forms, but it will necessarily include a vocabulary of terms and some specification of their meaning. This includes definitions and an indication of how concepts are inter-related which collectively impose a structure on the domain and constrain the possible interpretations of terms" [10].

3. Ontologies and Bayesian Networks

In this paragraph, we will describe how Bayesian Networks can be used "to map" and "to represent" an ontology. Bayesian Networks have been successfully used to model knowledge under conditions of uncertainty within expert systems, and methods have been developed from data combination and expert system knowledge in order to learn them [11]. Bayesian Networks represent a "hot" topic in the research field; the interested reader can find some interesting good surveys in [12][13]. In this paper a key role is played by the learning process of Bayesian Networks. It has two important advantages: firstly, it is easy to encode knowledge of an expert and such knowledge can be used to improve learning efficiency and accuracy. Secondly, nodes and arcs of the learned Bayesian network are recognizable links and causal relationships. So users can understand and exploit more easily the knowledge encoded in the representation. A Bayesian network is a graph-based model encoding the joint probability distribution of a set of random variables X = $\{X_1, \ldots, X_n\}$. It is composed by:

- A directed acyclic graph S (called structure) where each node is associated with one random variable

X_i and each arc represents the conditional dependence among the nodes that it joints,

- A set P of local probability distributions, each of which is associated with a random variable X_i and conditioned by the variables corresponding to the source nodes of the arcs entering the node with which X_i is associated. The lack of an arc between two nodes involves conditional independence. On the other hand, the presence of an arc from the node X_i to the node X_j represents that X_i is considered a direct cause of X_j. Given a structure S and the local probability distributions of each node $p(\bar{X}_i/\mathbf{Pa}_i)$, where $\mathbf{Pa}_i$ represents the set of parent nodes of X_i, the joint probability distribution $p(\mathbf{X})$ is obtained from: $p(X) = \prod_{i=1}^{n} p(X_i \mid Pa_i)$ and it is evident that the couple (S, P) encodes $p(\mathbf{X})$ unequivocally (on the hypothesis of conditional independence of the X_i given the $\mathbf{Pa_i}$) [11].

In order to build a Bayesian Network for a given set of variables, we have to define some arcs from the causal states to the other ones that represent their direct effects obtaining a network that accurately describes the conditional independence relations among the variables. The aim of this paper is the introduction of an algorithm, based on the formalism of the Bayesian networks, able to infer propaedeutic relationships among different subjects (in other terms the ontology) belonging to the knowledge domain of a university curricula. The first step of this algorithm is the introduction of a mapping between Ontology and Bayesian Network. In our ontology model, nodes represent the subjects belonging to the knowledge domain (the course) while the arcs mean a propaedeutic relationship among the nodes. We can map this ontology graph in a Bayesian Network in the following way: the Bayesian Network nodes model the subjects belonging to the course Knowledge Domain and the knowledge of subject by students while arcs in the same way mean the propaedeutic relationships among the nodes. Given the previous mapping strategy, our aim is to define the ontology used by teacher in his/her course. Obviously, we must define data type and data set for this approach. As previously said, student's answers to the evaluation tests represent a source of implicit evidence. In fact, teachers through the end-of-course evaluation tests not only assess student's knowledge for every subject, but describe the course ontology and outline the propaedeutic aspects that relate subjects each other. On the basis of these considerations, teachers have designed the final test of the first-level course on Computer Science at the Engineering Faculty of the University of Salerno and the final test of the first-level course on Introduction to Computer Science at the Languages Faculty of the University of Salerno. We must outline that this process was very long and hard for teachers. The result of this process is shown in Figure 1. Each node of the networks has two states and shows the probability that a generic learner knows the subject associated with the same node. We supposed that each node can assume only the following two states (random Bernoullian variable): state 'Yes': complete knowledge of the subject and state 'Not': total ignorance on the subject. The student level of knowledge could be evaluated on the basis of the answers given to the questions (a set of questions is proposed for each subject).

4. An Algorithm for Ontology Learning

As previously said, the teacher has difficulties sketching the relationships among the course subjects and their propaedeutic connections. A source of indirect evidence that can be employed for reconstructing a "a-posteriori" ontology can be represented by the end-of-course evaluation tests. On the basis of the ontology presented in Figure 1, some multiple-choice questionnaires have been realized. The previously described graph represents the ontologies, but can also be used as a Bayesian Network for the inference process. The student's level of knowledge is evaluated on the basis of the answers given to the questions. The presence of missing values, in other words the state of some variable can not be observable, has not been foreseen. This hypothesis can be obtained imposing that the student have answer to all the questions and considering a missing answer as a wrong one. Through a process of Bayesian inference conducted on the previously described networks, the candidate ontologies have been learned from data. The inference algorithm used in our experiments is the one called "*junction-tree*" introduced by Finn V. Jensen in [14]. For the inferential process, we have used data coming from five hundred questionnaires for the first ontology and three hundred questionnaires for the second and third ones. So, we have to estimate the strength of propaedeutic relationship between two arguments after the learning of the network. The presence of an arc between two nodes in the Bayesian network can be interpreted as the existence of a causality relationship between the variables associated to the same nodes. It is important to define a function able to evaluate this strength. For the nodes that belong to a Bayesian network a good dependence indicator is the cross-entropy function so defined:

$$C.E.(A,B) = \sum_{a,b} P(a,b) \log \frac{P(a,b)}{P(a)\,P(b)}$$

where A and B are nodes of the Bayesian network and a and b are the states of each node. According to cross entropy definition, we can say that A and B are independent if and only if C.E.(A, B) is equal to 0.

However, often we do not have the real probability distribution of the full network but only an empirical evaluation of it coming from data analysis. In this sence, it is incorrect to consider as condition of independence C.E.(A,B) = 0 and we can suppose A independent from B when C.E.(A,B)<e, where e>0 is an arbitrary threshold near to zero. The cross entropy function can also quantify the dependency weight between the nodes. In fact, an high value of C.E.(A,B) means a very strong preparatory relation between the two nodes.

In order to suppose that at least the father-child nodes links proposed by the teacher is correct, we submitted the data coming from the questionnaires to statistical tests, typical of Bayesian network structural learning algorithms that are able to establish from them the correct father-child nodes arrangement. This tests results confirmed that the arrangement proposed by teachers is correct. After this validation, we set as input to the Bayesian network the data coming from the questionnaires in order to obtain the probability values associated to the various states of the nodes. With these values we calculated the cross entropy values among all the single states of the net. Namely, the cross entropy has been calculated not only for the arcs proposed by the teacher but also for all the brother nodes. Figure 2 shows the obtained results. On the left side of the Figure, we can see the cross entropy values for the correct arcs that represent propaedeutic connection between two topics, while on the right side (after the blank column) we can see the cross entropy values for the incorrect arcs. In general, we can say that the arcs designated by teachers show a greater cross entropy values than other arcs so to confirm the teacher ontology design. We want also outline that in the case of ontology #1, an arc P(8|6) having a cross entropy value in the range of correct arcs is reported.

Given that examined data show a the existence of a significant value of cross entropy between these nodes the teacher, according this model, has to refine his original ontology proposal.

5. An assessment tool based on the ontology framework

In this section, we will describe in detail the architecture of an assessment tool based on ontology framework previously described. We designed our tool keeping in mind the main needs of students and teachers. In the first phase of the tool design, we pointed out the actors of the system and the use cases. We identified two actors in the system: Teachers and Students. Each of these figures has well defined role and tasks.

Teachers can design the reference ontology, describe the learning objects and the questions linked to the nodes of ontology. Teachers can also manage the reports of every student in order to better supervise the learning process.

Students can use tool in three different way: Exam, Normal test, Bayesian test.

In the Exam way, our tool arranges a classical final test exam according to the strategy of the teacher. In particular, teacher can choose the question's number for every subject and the scoring for every question. At the end of the exam the system produces a report analyzing the performance of student in every subject.

The normal test approach can be used during some module of the course. In particular, it can help student to better study the various learning objects.

The more interesting service offered by our tool is the Bayesian test. This service makes the most of the previously introduced matching between ontology and Bayesian Network.

In fact, the first step is the introduction of a mapping strategy between Ontology and Bayesian Network. As previously said in our ontology model nodes represent the subjects belonging to the knowledge domain of the course and the arcs mean a preparatory relationship among the nodes. According to the previously introduced method we can weigh the preparatory links among the ontology's nodes.

In this way the system can select a well defined set of questions associated to every network node. At the end of this first phase system, through the Bayesian approach infers what subjects the students knows better than others. In fact through the Bayesian analysis the system can measure the percentage of correct answer in a subject. In particular, it can predict the percentage of correct answer to a subject after a correct (or not) answer to questions related to propaedeutic subjects.

At this point it can apply various strategies: for example it can select and propose to the student the question with the smaller percentage of correct answer. At the end of Bayesian test a detailed report on the knowledge of student in the various subjects is sent to teacher and to student himself. Our tool proposes to the teacher a periodic report with the analysis of performances of various students in every subject. In this way, a teacher can easily understand where students need more help. At the end of Bayesian Test the system updates values of ontology's links according to the method introduced in this paper.

6. Conclusions

In this paper, we have presented a method for learning curricula ontologies. In particular, our approach is based on Bayesian networks. Thanks to their characteristics, these networks can be used to model and evaluate the conditional dependencies among the nodes of ontology on the basis of the data obtained from student tests. An experimental evaluation of the proposed method has been performed using real student data. We integrated the proposed method in a tool for the assessment of students during a learning process. This tool is based on the use of ontology and Bayesian Network. In particular through the matching between ontology and Bayesian Network our tool allow an effective tutoring and a better adaptation of learning process to demands of students. The assessment based on Bayesian approach allow as deeper analysis of student's knowledge.

7. References

[1] Neches R., Fikes R. E., Finin T., Gruber T. R., Senator T., Swartout W. R., *Enabling Technology for Knowledge Sharing*, AI Magazine, 12(3):36-56, 1991

[2] Uschold M., Gruninger M., *Ontologies: Principles, Methods and Applications*, Knowledge Engineering Review, volume 11, number 2, June 1992.

[3] Gomez-Perez, A.; Corcho, O., *Ontology languages for the Semantic Web, Intelligent Systems*, IEEE Volume 17, Issue 1, Jan/Feb 2002 Page(s):54 – 60

[4] Staab, S., Schnurr, H. P., Studer, R. Sure, Y., *Knowledge Processes and Ontologies,* IEEE Intelligent Systems 16(1), January/Febrary 2001

[5] Guarino, N., Welty, C., *Evaluating ontological decisions with OntoClean*, Communications of the ACM, 45(2), 2002

[6] Fernandez-Lopez, M., Gomez-Lopez, A., Pazos-Sierra, A., *Building a Chemical Ontology Using METHONTOLOGY and the Ontology Design Environment*, IEEE Intelligent Systems and their applications,1999

[7] Chandrasekaran, B., Josephson, J.R., Benjamins, R., *What are ontologies, and why do we need them?*, IEEE Intelligent Systems, 14(2), 1999

[8] Gruber, T.R, *Translation approach to portable ontology specification*, Knowledge Acquisition 5, 1993

[9] Maedche A., Staab S., *Ontology Learning for the Semantic Web*, IEEE Intelligent Systems, vol. 16 no. 2, Mar/Apr 2001, 72-79

[10] Uschold M., R. Jasper, *A Framework for Understanding and Classifying Ontology Applications*, IJCAI99 Workshop on Ontologies and Problem Solving Methods, Stockholm, 1999

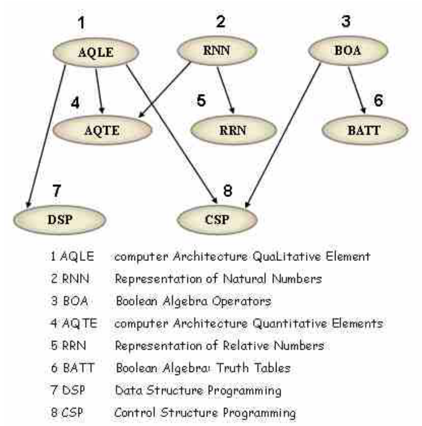
[11] Heckermann, D., *Bayesian Networks for Data Mining*, Journal of Knowledge Discovery and Data Mining 1(1), Kluwer Academic Publishers, 1997

[12] Singh, M., Valtorta, M., *Construction of Bayesian Network Structures from Data: a Brief Survey and an Efficient Algorithm,*. International Journal of Approximate Reasoning, 12, 1995

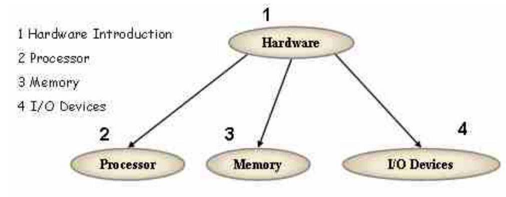
[13] Heckerman, D., Geiger, D., Chickering, D., L*earning Bayesian Network: the Combination of Knowledge and Statistical Data*, Machine Learning, 20, 1995

[14] Jensen, F., *An Introduction to Bayesian Networks*, Springer – Verlag, New York, 1998

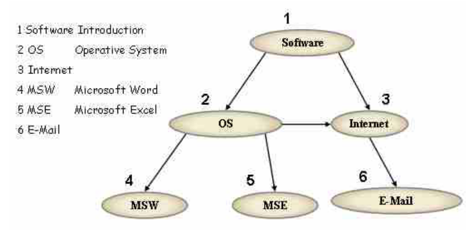
8. Figures



Ontology #1

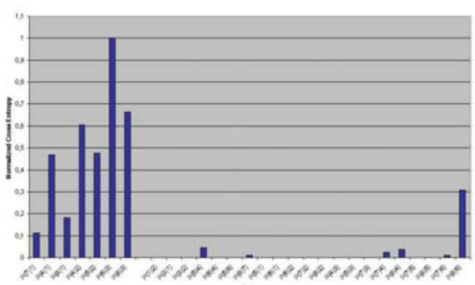


Ontology #2

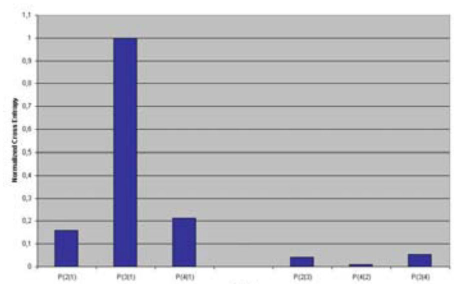


Ontology #3

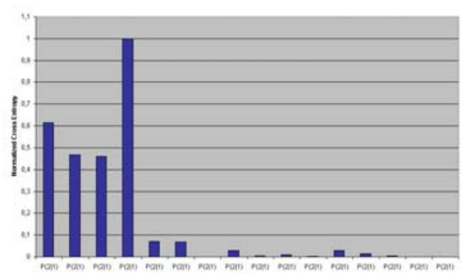
Figure 1 Proposed ontology for the first-level course on Computer Science (Ontology #1) and Introduction to Computer Science (Ontology #2 and Ontology #3)



Ontology #1



Ontology #2



Ontology #3

Figure 2 Obtained Results on reference set ontologies

TIME MANAGEMENT FOR SENIOR CITIZEN CARE

Shi-Kuo Chang, Suresh Rangan and Yue Zhang
Department of Computer Science, University of Pittsburgh
Pittsburgh, PA 15260 USA (chang@cs.pitt.edu)

Abstract: *The chronobot allows a group of people to exchange time and knowledge. This paper describes the basic concept of the chronobot/virtual classroom (CVC) system, time/knowledge exchange, time management and scheduling for an application of the CVC system to senior citizens care.*

1. Introduction

We describe a distributed multimedia system for storing and borrowing time. Using the chronobot [1] one can borrow time from someone and return time to the same person or someone else.

The main idea of the chronobot is flexible allocation of one's time and knowledge to achieve the best match for time and knowledge exchange among different agents through mediated negotiation [2]. Such idea is gaining popularity in recent years due to advances in information technology. For example at Nashoba Valley Medical Center registered nurses can bid on working shifts that have openings. The chronobot will make this possible at the personal level in all works of life.

The chronobot can be very useful to senior citizens who can use the chronobot to request services from professional nurses and volunteer workers. They can also use a virtual classroom to attend classes, learn new skills, and offer their knowledge to others by teaching classes. Therefore they can use the chronobot and the virtual classroom to exchange time and knowledge.

2. Characteristics of the Chronobot

As mentioned above the chronobot is designed for time and knowledge exchange. As illustrated in Figure 1 the chronobot and the virtual classroom together form the CVC system, which is a distributed multimedia system with the following characteristics:

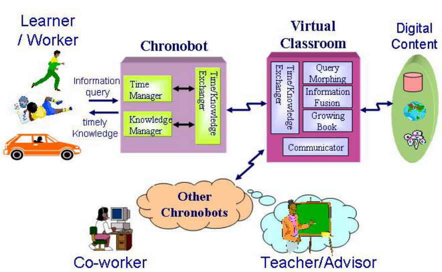


Figure 1. Chronobot and Virtual Classroom.

1. The chronobot is a **time manager**. But it is not an ordinary time manager. It can be used to **manage not only one's own time, but also other people's time through time/knowledge exchange**. This is very important, because the unique concept of the chronobot is that **time and knowledge are exchangeable**. The chronobot can manage not only the present, but also the past and the future through suitable time/knowledge exchange protocols.

2. The chronobot is also a **knowledge manager**. It can be used to store knowledge, organize knowledge, retrieve knowledge and **perform information fusion to produce new knowledge**. Information fusion is the key concept. Without information fusion, the chronobot will not be as effective in managing knowledge and saving time.

3. For e-learning and distance education applications, the chronobot offers a versatile **virtual classroom** that combines the functions of chat room, white board, multimedia display. The learning materials, references and related information become a

continuously expanding knowledge source or a Growing Book. Evolutionary query processing provides query morphing and information fusion capabilities to fully utilize this ever-changing knowledge source.

4. Utilizing the time manager, the knowledge manager and the virtual classroom, the **chronobot can interactively provide timely knowledge to the end user**. This is the key point of the chronobot.

3. Time/Knowledge Exchange System

The methodology for time and knowledge exchange essentially consists of the following five steps:

(1) **Identify** a slice of time or knowledge for exchange.
(2) **Search** for exchange partner or partners.
(3) **Perform** time or knowledge exchange through bidding and negotiation.
(4) **Manage** the exchanged slice of time or knowledge.
(5) **Evaluate** the results.

A *time/knowledge exchange system* consists of agents such as people and chronobots. Each agent *u* can be defined as follows:

u=(*user-model*,
relation-search-mechanism,
negotiation-protocol, *time-schedule*,
knowledge-base, operations)

The chronobot agent *u* is defined as:
u=(P_u, RI_u, $Nego\text{-}QoB_u$, TS_u, KB_u, OP_u)
where P_u is user profile,
RI_u is relational index,
$Nego\text{-}QoB_u$ is negotiation protocol according to quality of bid,
TS_u is time schedule,
KB_u is ontological knowledge-base
OP_u is set of operations for agent *u*

The user profile is used to specify a user's preferences, skills, etc. The chronobot stores relational information in a relational index, which is searched to find exchange partner(s). The chronobot then supports bidding and negotiation according to quality of bid. The details are explained in [2]. In the next section, time schedule, time management and the use of an ontological knowledge-base will be explained in more detail.

4. Time and Knowledge Management

Time management for the chronobot differs from traditional time management in that all three ingredients - time, space and tasks - are considered in determining a time schedule. Therefore time management and knowledge management are inseparable. In this section we introduce the basic concepts.

A *time schedule* TS_v of an agent *v* is an assignment from the time line T to the state space S where each state is a pair (location, task). The agent may be a person, a project, a corporation, an institution and so on. In particular, a *life time schedule* LTS_u, or simply a *life*, is a time schedule of a person *u* where $LTS_u(t)$ = (no-where, no-task) if $t \le t_{u\text{-birth}}$ or $t \ge t_{u\text{-death}}$.

Two locations are compatible if either both are real locations and they are either identical or their distance is within a predefined threshold, or one location is a virtual location and they are compatible according to a location compatibility matrix, or both locations are virtual. Two tasks are compatible if $task_1$ and $task_2$ are compatible according to a task compatibility ontology. Two states s_1 and s_2 are *compatible* if both locations and tasks are compatible.

A time schedule TS_v is *supported* by a life time schedule LTS_u in the interval $[t_a, t_b]$ if $TS_v(t)$ is compatible with $LTS_u(t)$ for any t in $[t_a, t_b]$. A time schedule TS_v is *feasible* in the interval $[t_a, t_b]$ with the support of G if for any sub-interval $[t_c, t_d]$ of $[t_a, t_b]$ there exists a *u* in G such that TS_v is *supported* by a life time schedule LTS_u in $[t_c, t_d]$. In practice the sub-intervals are predefined time periods. Finally a *life model* LM is an approach to generate a certain type of feasible life time schedules.

The Knowledge Manager checks the compatibility of time schedules and life time schedules with respect to a knowledge base containing information on location compatibility and task compatibility. A task may also have certain special characteristics such as: must-be-done-by-self, can-be-done-by-others, must-be-carried-out-at-certain-location, can-be-carried-out-anywhere, can-be-carried-out-in-virtual-locations and so on. Such task characteristics are also stored in the knowledge base and utilized by the Knowledge Manager in determining compatibility.

Supported by the Knowledge Manager, the Time Manager checks whether a time schedule is feasible. If it is not feasible, the Time Manager attempts to revise the time schedule and *out-source* certain tasks by revising the life time schedules of agents in G to make the revised schedules feasible.

When a time schedule is infeasible, it is sometimes possible to make it feasible by revising the schedule or absorbing certain tasks in the free time of a person. Therefore it is not always necessary to out-source. Sometimes a task has explicit constraints so that it can only be done by oneself.

The Knowledge Manager makes sure certain tasks are compatible with certain persons by checking an ontology. The Time Manager can then automatically place a bid if the user has previously authorized the Time Manager to do so, or at least notify the user.

The following is an example of time/knowledge management using the concepts introduced above. The following small ontology for some related keywords can be part of a much larger ontology. However for practicality we may want to refrain from using any ontology more complicated than a simple hierarchy:

```
Health care:
   Physical therapy:
       Leg therapy
       Arm therapy
       …..
```

George is a senior citizen who is also a patient because of a stroke, and his Life Time Schedule LTS for today is:

1-2 pm	2-3 pm	3-4 pm	4-5 pm
*	Home	Gym	Gym
Caligraphy	Leg therapy	Jog	Jog

In George's schedule the asterisk indicates virtual space, i.e., George is willing to exchange time with others to teach caligraphy in a virtual space. Mary is a nurse and her Life Time Schedule LTS is:

1-2 pm	2-3 pm
Any location	Any location
Physical therapy	Physical therapy

According to the above ontology, "physical therapy" and "leg therapy" are compatible. Therefore the two LTSs are also compatible, and Mary's LTS can support George's LTS. If Mary places a bid in the bidding room, the Bid Manager may grant her the bid provided that she is the most suitable bidder according to its QoB calculations.

If Mary has not entered a bid in the bidding room, George may still be able to find him using a Searcher supported by the Relational Index, which may contain a record such as the following due to prior interactions between Mary and other users:

```
[physical therapy; Mary; 26]
```

5. Scheduling for Senior Citizen Care

Many different considerations may enter into time management, depending upon the specific application. Continuing the application example described in the previous section, when we consider the matching of time schedule between nurses and patients, it becomes necessary to consider the traveling time and scheduling of nurses.

As illustrated in Figure 4, a nurse may try to serve several patients in different locations. A nurse must serve a patient during the time period when the patient is ready for being served. Moreover a nurse can only serve a maximum number of patients in one day.

Each nurse or patient has a time schedule specifying when he or she should do what. For example a patient's time schedule is shown below in Figure 5. This patient can be served between 10:10am and 10:40am.

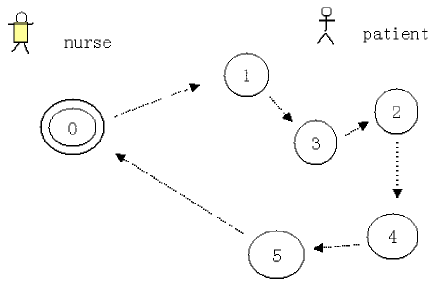


Figure 4. The service route of a nurse.

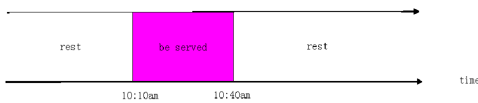


Figure 5. The time schedule of a patient.

Similarly, Figure 6 shows the time schedule of a nurse. The nurse can only work during 8:00am and 12:00pm or 1:00pm and 6:00pm. The rest period of a patient/nurse must also be respected.

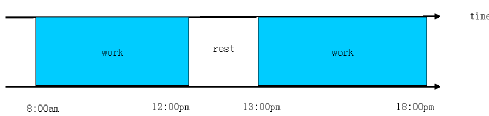


Figure 6. The time schedule of a nurse.

The time management algorithm tries to minimize the number of nurses needed to service a given number of patients with time scheduling constraints. For a given nurse, the time management algorithm then tries to minimize the travel times and maximize the number of patients served by this nurse.

Two time management algorithms were developed: a heuristic algorithm *HA* and a genetic algorithm *GA*.

The heuristic algorithm *HA* uses a heuristic function H to estimate the goodness of a (partial) solution [5]. For a nurse *n*, *HA* attempts to find a patient *p* that minimizes H(*n*, *p*), defined as current_time(*n*) + travel_time(*n*, *p*) + serve_time(*p*), as illustrated in Figure 7.

When deciding whether to serve a patient, a nurse will first travel to the patient's location and then wait for the patient to be served. After serving, the nurse is available to serve other patients.

Algorithm *HA*

For each nurse *n*

Step 1. consider each patient *p*, if no more patient can be served, exit.

Step 2. if one or more patients can be served by this nurse *n*, choose the patient *p* leading to the smallest H(*n*, *p*) value to serve.

Step 3. if the number of patients the nurse has served reaches the maximum number the nurse can serve in one day, exit. Otherwise go to step 1.

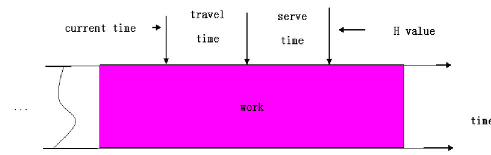


Figure 7. The heuristic function H(*n*, *p*).

Algorithm *GA* is based upon the genetic algorithm [3, 4]. However the genetic operators are modified to include crossover, mutation and deletion in order to handle nurse scheduling for senior citizen care.

Algorithm *GA*

Step 1. From the input list of Location and Service Times, Create an Ordered Services Times based on Segment Start Time and Segment End Time.

Step 2. Generate base chromosomes. Each service time represents a gene. Base chromosomes are created with the first gene of the chromosome picked at random from the ordered service times collection. The remaining genes of the chromosomes are formed as per the order in the ordered service times collection. A few basic rules are applied to evaluate the formation of initial population. One such rule is those service times should not overlap. A calculated number of base chromosomes are formed in order to generate the required number of initial population.

Step 3. Each base chromosome is taken and applied the GA mechanism called *Deletion*. This is exactly the opposite of the GA mechanism called *Insertion*. Depending on the length of the base chromosome and the required length of the chromosome, the number of genes to be deleted is determined. Keeping the order in place, randomly delete the genes to get a valid chromosome. Repeat the process to obtain more chromosomes. The same step is executed for all the base chromosomes until the required population size is reached.

Step 4. Apply fitness parameters and travel times constraints. Start with the fixed number of generations with the maximum possible length of the chromosome (say 5) that will be the schedule. Try to possibly fit in as much as service times and locations as possible. Repeat the steps taking a *step-down* approach of one lesser length of the chromosomes (say 4..3..2). This will ensure that the schedule covering all the locations/service times will always be found.

6. Experimental Results

In this section we describe the experimental results of nurse scheduling using algorithm *HA* and algorithm *GA*. In problems 1 and 3 each senior citizen has a different home location. In problems 2 and 4 some of them share the same home. The senior citizens, their home locations and periods to receive service are as follows:

0	(13,44)	[9:30,9:50]
1	(45,32)	[14:10,14:40]
2	(6,89)	[10:10,10:20]
3*	(6,89)	[10:20,10:30]
4*	(6,89)	[10:30,10:40]
5	(55,71)	[9:40,10:00]
6	(85,21)	[9:30,9:50]
7	(28,60)	[11:10,11:50]
8	(19,4)	[15:20,15:40]
9*	(19,4)	[15:20,15:40]
10*	(19,4)	[15:20,15:40]
11*	(19,4)	[15:30,15:40]
12*	(19,4)	[15:30,15:40]
13*	(19,4)	[15:30,15:40]
14*	(19,4)	[15:40,16:00]
15*	(19,4)	[15:40,16:00]
16*	(19,4)	[15:40,16:00]
17*	(19,4)	[15:40,16:00]
18	(76,11)	[17:10,17:30]
19	(32,49)	[13:20,13:30]
20	(91,3)	[15:10,15:40]
21	(34,38)	[11:00,11;20]
22	(90,57)	[14:00,14:20]
23	(22,75)	[16:20,16:50]
24	(71,96)	[8:50,9:10]
25	(66,16)	[10:20,10:50]
26*	(66,16)	[10:20,10:50]
27*	(66,16)	[15:50,16;10]
28*	(66,16)	[15:50,16;10]
29*	(66,16)	[15:50,16;10]

In the above an asterisk means the patient only appears in problems 2 and 4. All nurses have working periods [8:00,12:00] and [13:00,18:00]. In problems 1 and 2 all nurses start from a single service station at location (50,50). In problems 3 and 4 service station #1 at (18, 2) has 3 nurses, station #2 at (55,60) has 4 nurses, station #3 at (81, 70) has 10 nurses and station #4 at (91, 21) has 6 nurses. A nurse can serve at most 5 patients per day. A nurse's traveling speed is 2 space_unit/time_unit. Therefore to travel from (50, 50) to (19, 4) takes 55.47/2 = 28 time_units.

The experimental results for problem 1 is illustrated in figures 8(a) and 8(b). *HA* found the optimal number of nurses (4) very quickly, but *GA* is able to reduce the average travel times of the nurses.

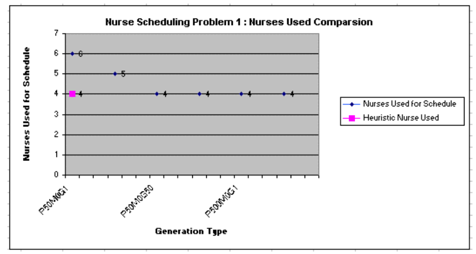


Figure 8(a). Nurses used for problem 1.

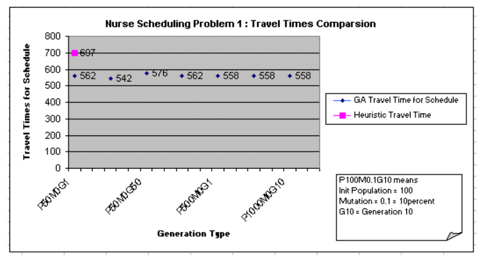


Figure 8(b). Travel times for problem 1.

Fig. 8(c) and 8(d) are both visualizations for *GA*. In Figure 8(c) each colored itinerary represents a nurse's travel schedule that always starts and ends at the same nurse's location indicated by blue numbers. The red numbers indicate senior citizen's locations.

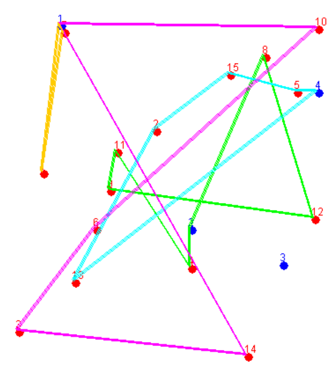


Figure 8(c). Nurses' itineraries.

In Figure 8(d) the nurses' travel schedules plotted by a Java applet are shown in different colors. The filled segments are service times and the unfilled segments are travel times.

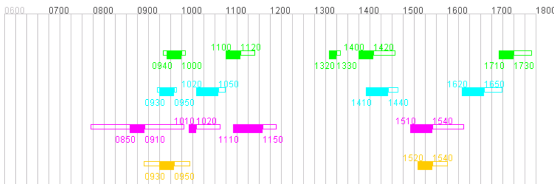


Figure 8(d). Service times and travel times.

The experimental results for problem 2 are illustrated in figure 9(a) and 9(b). Algorithm *GA* reduces number of nurses from 11 to 8 and also average travel times.

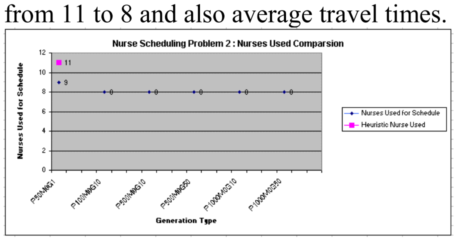


Figure 9(a). Nurses used for problem 2.

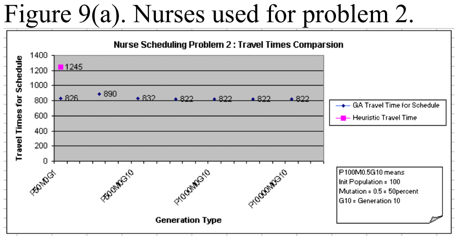


Figure 9(b). Travel times for problem 2.

The experimental results for problem 3 is illustrated in Figure 10. *GA* reduces average travel times, both *GA* and *HA* use 4 nurses.

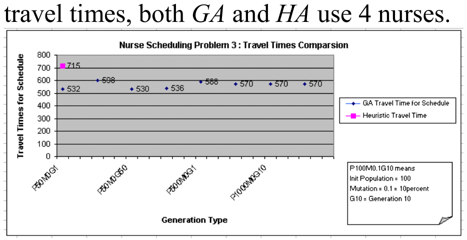


Figure 10. Travel times for problem 3.

The experimental results for problem 4 is illustrated in Figure 11. *GA* reduces average travel times, and nurses used from 11 to 8.

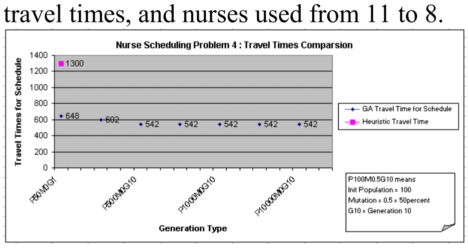


Figure 11. Travel times for problem 4.

7. Implementation and Discussion

The nurse scheduling *GA* algorithm is integrated into the experimental chronobot system for senior-citizen-care as servlets, and a visualization applet is used to display the nurses' itineraries and schedules (see figures 8(c) and 8(d) above).

The heuristic algorithm *HA* and the genetic algorithm *GA* each has strong points. *HA* has time complexity O(p) where p is the number of patients. It can find a good solution quickly. *GA* has several unique features not commonly found in other genetic algorithms, including the *crossover* operation and the *step-down* approach. It is time consuming but yields better solutions. They can be combined into a hybrid algorithm *HYA* via problem decomposition: *HA* is used to provide initial solutions for *GA*.

As discussed in Section 4, time management involves many aspects. We also need to take into consideration knowledge-based constraints, which will be incorporated into the hybrid algorithm we are working on.

References:

[1] Shi-Kuo Chang, *Nocturne*, a science fiction short story first published in Chinese in 1980; electronic version in English is available at: http://jupiter.ksi.edu/~changsk/chronobot/nocturne.doc.

[2] Shi-Kuo Chang, "A Chronobot for Time and Knowledge Exchange in e-Learning", Int'l Journal of Distance Education Technologies *JDET*, April 2005.

[3] Ann S K Kwan, S. K. Raymond and Anthony Wren, "Driver scheduling using genetic algorithms with embedded combinatorial traits", in: N H M Wilson (ed.), *Computer-Aided Transit Scheduling*, pp. 81-102 Springer-Verlag, 1999.

[4] Tadahiko Murata and Mitsuo Gen: "Performance Evaluation of Solution-Based GA and Rule-Based GA for scheduling Problems", Proc. of The Fourth Asian Fuzzy System Symposium, Tsukuba, Japan, May 31-June 3, 2000, 696-699.

[5] T.C. Raymond: "Heuristic Algorithm for the Traveling-salesman Problem", IBM journal of Research and Development, Vol. 13, No. 4, 1969, 400-407.

Integrating e-Business and e-Learning Processes

Andrea De Lucia, Rita Francese, Giuseppe Scanniello, Genoveffa Tortora

Dipartimento di Matematica e Informatica, University of Salerno
Via Ponte Don Melillo, 84084 Fisciano (SA), ITALY
{adelucia, francese, gscanniello, tortora}@unisa.it

Abstract

In this paper we propose a system integrating a distributed Workflow Management System and a Learning Management System. The Workflow Management component offers a visual environment based on an extension of UML activity diagrams that allows to graphically design a process model and to visually monitor its enactment. Similarly, an extension of UML activity diagrams is used by the Learning Management component for defining e-learning processes. The proposed system enables to associate activities of the business process with e-learning processes. In this way, a human resource involved in an e-business process can bridge the gap while performing an activity.

1 Introduction

The definition and the management of business processes are considered a relevant issue to support organisations in their activities. Thus, many organisations have been changing their business processes to keep competitive in the global market. Workflow management represents an emerging technology to improve the process performance in a cooperative environment.

Workflow Management Systems (WfMSs) enables process automation through the integration, the coordination, and communication of both human and automatic task of business processes. WfMSs provide a Process Definition Language (PDL) for modelling business processes. Several PDLs based on several formalisms has been proposed in literature, e.g. [3], [15], [12]. Recently, some authors suggested to exploit the Unified Modelling Language (UML) proposed by the Object Management Group [17] to model business processes [2][9][11][13][16]. Generally, WfMSs do not support the knowledge improvement of human resources by learning on the job and learning on demand. Thus, the human resources can only trust on their knowledge to accomplish activities belonging to business processes or use external e-learning systems.

In this paper we propose a system integrating a distributed WfMS with the Learning Management System (LMS) described from Costagliola *et al* in [6]. In particular, the process definition tool of the WfMS offers a visual environment based on an extension of UML activity diagrams with object flows that allows to graphically design a process model and to visually monitor its enactment. Since UML does not have a well defined operational semantics and is not executable, we had to make the process model executable by appropriately enriching the syntax and semantics of UML activity diagrams. The architecture of the proposed WfMS is based on Web Services to manage and enact distributed business processes. The proposed system associates activities of the business process with e-learning processes also modelled by using an extension of UML activity diagrams. The common UML based notation for both types of processes is a key for this integration. The e-learning processes also allow to adaptively accessing the knowledge contents considering the role, the background and the tasks to be performed by each human resource. The learning process is designed using assessment rules aimed at evaluating the actual knowledge of the human resource referring to a given role before the business activity is enacted.

The remainder of this paper is organised as follows. Section 2 discusses related work, while the UML based visual notation to define e-business and e-learning processes are proposed in Section 3 and Section 4, respectively. The WfMS architecture is presented in Section 5, while final remarks and future work are discussed in Section 6.

2 Related Work

2.1 Workflow Management System

Many Workflow Management Systems (WfMSs) [21] have been developed in the last decades to provide support for the modelling, improvement, and automation of business and industrial engineering processes

[7][10][20], including software processes [12][14].

In general, several PDLs based on different formalisms have been proposed. For example, Aversano *et al.* [3] and Loops *et al.* [15] use event-condition-action mechanisms, while a graph rewriting mechanism is proposed by Heimann *et al.* [12]. On the other hand, several researchers have recently proposed to adopt UML [17] for representing business processes [9][11][13][16] Thus, to model business processes at a very high level [15][11][16] most approaches proposed in the literature use UML, and in some cases manually turn the UML model into the PDL of a specific workflow management system [2][15].

Recently, some research efforts are being made to add a well defined operational semantics to UML notations to automatically derive executable process models [13]. In particular, Jager *et al.* [13] propose the UML class diagrams and the state diagrams to model the structure and the behaviour of a process. Dynamic semantics is added to translate the UML models into dynamic activity nets, which dynamically evolve during enactment based on graph rewriting systems [12]. Differently, Di Nitto *et al.* [9] propose a subset of UML diagrams, including UML activity diagrams with object flow to model the control and data flow, class diagrams to model structural properties of the process, and state diagrams to model business processes as well as the behaviour of activities. The XMI standard representation of these models produced using a UML CASE tool can then be translated into an executable process description for the OPSS Workflow Management System [7]. De Lucia *et al.* [8] have presented a case study of mapping UML activity diagrams with object flow on the process definition language of the GENESIS environment [3].

2.2 Learning Management Systems

The e-learning research field proposes a good number of academic and commercial tools assisting the instructional designer during the analysis, the design, the implementation, and the delivery of instruction via the Web [4]. In particular, Vrasidas [19] presents and discusses a system to develop hypermedia approaches as part of courses and learning environments delivered via the Web. He details the structuring of information, branching and interactivity, user interface, and navigation through Web based distance courses. In the AIMS [1] Project a Theoretical Framework is described, which distinguishes the knowledge domain editing from the course editing. Cooke and Thomson [18] propose the APHID method to support designers during the course definition by instructional patterns. Differently from these approaches, an extension of the UML Activity Diagrams [17] to define adaptive learning processes based on asynchronous activities has been proposed in [6].

3 Modelling Business Processes using UML activity diagrams

Since UML activity diagrams provide an intuitive and easy to learn PDL, we use an extension of the activity diagrams with object-flow [17] to model business processes. UML activity diagrams are a particular variation of UML state diagrams where states represents actions (or activities) and are modelled by rounded rectangles and transitions between states depicted as solid arrows model the control flow between two activities. Activity diagrams have been enhanced with object flow to depict the data flow between activities.

Although structural properties and relations between process elements, such as artefacts, activities, and roles, could be specified using other UML diagrams [9]. To avoid the use of too many diagrams we did not include them in the proposed PDL. Indeed, a process modelling tool can provide other features to model these aspects, such as forms or simpler graphical notations (e.g., organisational charts for the roles). Some approaches [9][13] use state diagrams for modelling the internal behaviour of activities. In our approach, activities may be interactive or automatic depending on if they are performed by humans or by a tool, respectively. We can use the actor symbol and the cog symbol in the left-higher corner of the activity symbol to denote interactive and automatic activities, respectively. As an example of human activity see the *OrderReceiver* activity in Figure 1A. When no differently specified, the default value of the activity type is automatic. Unlike other WfMSs, for interactive activities we do not provide a visual notation to model them. Rather, their user interface is automatically generated from the definition of the input and output objects of the interactive activity. For each identified role involved in activity enactment a suited learning process may be defined. Automatic activities are associated to Web Services using Web Service Description Language (WSDL) specification and executed on remote machines.

We needed to suitably enrich the syntax and semantics of UML activity diagrams to enable the specification and execution of particular aspects of a distributed business process. In particular, distributed processes are organised in a hierarchical way and modelled as sub-processes in UML activity diagrams (see Figure 1 for an example). A compound activity is depicted as an activity with the addition of an icon in the right lower corner denoting a nested activity diagram, see the activity *Bill Handler* in Figure 1A. In particular, Figure 1A shows the *Order*

Handling process and Figure 1B the sub-process associated to the composed activity *Production Handler*. The input artefacts are transferred to the sub-process as if they were produced by the start activity. Similarly, all the output artefacts are transferred to the higher level process by assigning them to the end activity.

In UML activity diagrams it is possible to specify whether concurrent instances of the same activity can be executed and in this case their multiplicity. We can use the *(*)* symbol (multiplicity marker) in the right-higher corner of the activity symbol, as shown in the activity *Check Stock* in Figure 1A to denote that multiple instances of the activity can be executed. This symbol indicates that the number of instances is unknown during the process modelling and will be determined at run-time. Differently, when the exact number of instances is known the multiplicity can be specified.

In UML activity diagrams with object flow, a specific object with a name is exchanged between two activities. On the other hand we need to provide more flexibility, in that an undefined number of artefacts of a given type can be produced by the source activity and consumed by the target activity. Therefore, we use for artefact names the typical UML object notation *artefact-name:artefact-type* in case the multiplicity of the artefact is one. Whenever the multiplicity of the artefact type is undefined (many), we use the notation*: artefact-type*, without specifying the name of the artefact (see the *:Order* object in Figure 1A as an example).

In case of a transition between a source activity and a workflow object we need to define a way to distinguish on one end whether a produced artefact is newly created or updated by an activity and on the other end whether an input artefact has also to be updated by an activity.

In UML the same object can appear more than once in the diagram to denote updates of the object with a different state placed in brackets and appended to the name of the object. In case of artefacts with multiplicity one, the UML visual syntax is enough to distinguish input from updated and newly created artefacts: indeed an artefact will be updated only if it is both input and output for an activity. In case of artefacts with undefined multiplicity we need a different way to make such a distinction. In particular, we decided to label the arrows with stereotypes as shown in Figure 1: the <<*update*>> label on an incoming data flow edge means that the activity also updates the input artefact (there is no need of specifying the label in case the input artefacts are not modified), as in case of the activity *Production Admin* in Figure 1B. Output artefacts of an activity are considered newly created if no stereotype is specified on the edge (see *:ProductionPlainReport* artefact in Figure 1B). In case the activity also updates artefacts of the same type, these artefacts will also be considered in the outgoing data flow transition.

The *Prodution Admin* activity in Figure 1B depicts the case where output artefacts of type*: ProductionPlanReport* of activity *Production Plan* are only the result of updating the corresponding input artefacts of the same type, without any new creation of artefacts of that type. In this case the outgoing edge of the activity is labelled with the stereotype <<*update-only*>> and the incoming edge must be labelled with the stereotype <<*update*>>.

To model the synchronisation between two or more transitions UML synchronisation bars can be used. We associate to the synchronization bar a different operational semantic depending on the synchronisation type. We have identified different synchronisation types that considering the synchronisation problems we have encountered in modelling workflow processes. The

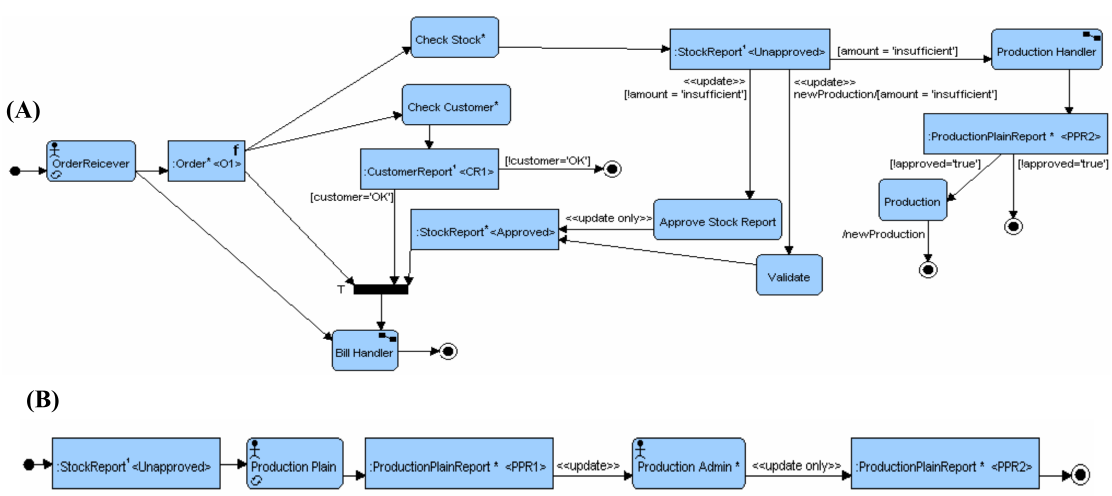


Figure 1. Process and sub-process modelling

synchronisation type can be depicted as a label on the left of the UML synchronisation bar, as shown in Figure 1A, where the synchronisation bar is labelled *T*.

As an example, the traceability synchronisation type is used when activities or artefacts with multiplicity greater than one instantiate several artefacts (or activities) of the same type that have to be synchronised with other artefacts or activities. We need a way to distinguish the activities and the artefacts belonging to the same group. To this aim, we identify a point of the process, the traceability fire, where a new traceability identifier is generated. A traceability fire is depicted by adding *f* as subscript to the name of the node, as in the artefact type *Order* in Figure 1A. Here the *OderReceiver* activity produces several artefacts of type *Order* having a traceability fire assigned to them. Whenever a new instance of the artefact is produced a new traceability identifier is associated to it and the two instances of *Check Stock* and *Get Customer* having the same traceability identifier are activated. The edge having the traceability join as source is started when the join node receives as input three objects of type :*Order*, *:CustomerReport* and *:StockReport* with the same traceability identifier.

4 Modelling e-learning Processes Using UML Activity Diagram

The system enables to associate e-learning processes to human activities. The e-learning processes are also modelled by using a visual formalism, an extension of UML activity diagrams. This formalism [6] assembles predefined didactic activities to define e-learning processes, which are broken down and structured into a hierarchy from smaller, lower order blocks of material to higher, more complicated levels of learning. In particular, three different granularity levels referring to the size of knowledge contents were identified. Therefore, the authors proposed a hierarchy of three visual languages to be employed during the different phases of the learning design process.

The first proposed visual language is the Learning Activity Diagrams (LAD). LAD extends the Activity Diagrams of UML [17] in order to enable the modelling of the learning process based on asynchronous learning activities. LAD was conceived to specify the relationships among e-learning activities, which can be further refined by using a visual sentence belonging to either the Self-Consistent Learning Object (SCLO) language or the Test Maker Language (TML). SCLO is a special case of state transition diagrams, and enables the instruction designer to define learning content objects. On the other side, TML extends state diagrams to enable the design of assessment and self-assessment tests. It lets us describe tests that adapt themselves to learner's answers. Tests play an important role in such an approach. Indeed, they allow defining learning processes adapting themselves to learner performance. Details on the SCLO and TML languages can be found in [6].

In general, LAD allows describing educational materials, dependences, and assessment rules. Material dependences allow varying the degree of control over the order in which the learners must explore the materials spread in SCLO objects. Moreover, using the results of the self-assessment tests the flow of the learning process is adapted to the learning performance. For example, before taking up a course we can define a test whose result is used to properly adapt the learning process.

An example of LAD sentence is shown in Figure 2. It which describes a learning process for a Production Manager associated to the activity *Production Plan* of Figure 1. The rectangles represent SCLO objects refined by using the SCLO visual language, while self-assessment activities are represented by diamonds. A rectangle represents a state of the learning process that is left when the associated learning object is completely executed. SCLOs can be also refined through LAD sentences associating e-learning sub-process. When this happens the icon shows a nested structure, as in case of business sub-processes described in the previous section. The considered process has been structured into three swimlanes depending on the manager competences, which are evaluated through the self-assessment activity on the top of the sentence. Therefore, the manager knowledge is considered *Basic* when a score less than 50% is obtained. It means that he/she has to study in depth the contents presented within the *Basic* swimlane. This swimlane includes didactic contents aimed at providing detailed knowledge of production planning. Differently, when the test result is between 51% and 90% the knowledge contents will be a resume of the *Basic* swimlane teaching materials. Finally, the *Advanced* swimlane presents more general contents to an expert manager. It is worth noting that the *Basic* swimlane proposes three SCLOs that can be consumed in parallel. A self assessment test follows the activity *Case Study* to impose a manager with deficiencies to revise the *MRP 1*, *MRP 2*, and *Case Study*contents.

5 The Integrated WfMS and LMS architecture

The architecture of the system follows the model proposed by the Workflow Management Coalition [21]. Figure 3 shows the architecture in terms of nodes and software components.

The *Process Definition Tool* enables the definition of the process according to the UML based PDL described above. This is stored in the database in terms of the process components (activities, artefacts, transitions, etc.). It also uses the Authoring tool [6] features (implementing the above visual language hierarchy) to associate an e-learning processes to an activity. The aim is to provide training on the job to a participant of the project, considering his/her role, his/her skills and the tasks that he/she has to perform to contribute to the activity enactment. In particular, the Authoring tool components allows associating, defining, designing, updating, and generating e-learning processes, which are successively deployed and stored in the database through the *LMS* software component.

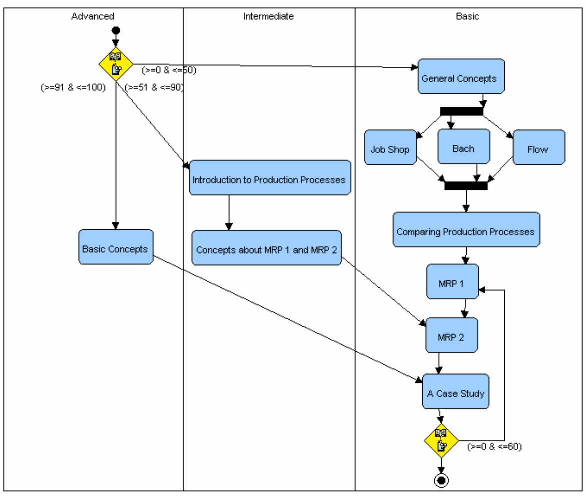


Figure 2. A LAD sentence

The *Workflow Engine* is a component whose main task is to manage the execution of process instances through the enactment rules specified in the process model. The Workflow Engine is in charge of instantiating both types of activities, automatic and interactive. Besides instantiating an automatic activity, the workflow engine needs to identify the corresponding Web Service. To this aim the *Activity Handler* component is used. This is a component that is in charge to identify and select the Web Service corresponding to the definition of the automatic activity. Once this has been identified and registered in the UDDI registry, the Activity Handler creates the instance of the automatic activity in the database and associates the URL of the corresponding Web Service. The Activity Handler is in charge of monitoring the evolution of the activity, storing the data returned by the corresponding Web Service and notifying the *Event Handler* (a component of the Workflow Engine) of events concerning the activity, such as its termination or the production of an object. These events are stored in the database and later processed by the Workflow Engine according to the workflow enactment rules.

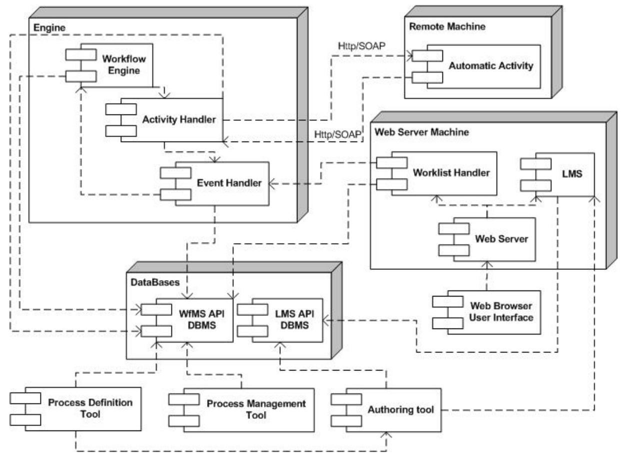


Figure 3. The architecture

Compound activities corresponding to sub-processes are handled as automatic activities. Whenever a compound activity has to be instantiated, the Activity Handler identifies the Workflow Management System that is in charge of executing the sub-process. The latter maintains the model of the sub-process and the data concerning the instance execution in the local database and only events concerning the change of state of the process instance or the production of output objects are monitored by the Activity Handler and notified to the Event Handler of the higher level process.

Interactive activities are simply created in the database by the workflow engine and are managed by the *Worklist Handler*, a component developed using Web technologies, which optimises the assignment of human resources by considering the effort needed to accomplish each activity. The graphical interface of an interactive activity is generated through the Worklist Handler, and is presented through Web browser on the client machine. The Worklist Handler provides information on the activities that can be executed, the corresponding state (such as running, suspended, or terminated), and its input and output objects. It also allows a human resource to acquire particular competences he/she needs before performing an activity by using the features offered by the integrated LMS for the delivery of e-learning process and the management of learning progress. The Worklist Handler stores the objects produced by the interactive activities into the database and notifies events concerning interactive activities to the Event Handler.

Finally, the *Process Management* tool is the component which handles the process execution. It allows starting a process, monitoring its state during enactment and managing deviations from the process model in case of unforeseen situations [5].

6 Conclusion and Future Work

After a brief description of the main characteristics of a distributed WfMS and of a LMS, both based on an extension of the UML activity diagrams, we have presented the system integrating both the considered components. We distinguished between automatic and interactive activities. Automatic activities are managed as Web Services, whilst the interfaces of interactive activities are automatically generated starting from the specification of the input and output artefacts. The proposed system enables the business manager to define and enact interactive activities supporting the training on the job and learning on demand. To this end e-learning processes are associated to human activities. Thus, human resources acquire particular missing competences before accomplishing the activities of the process.

Future work will be devoted to further support the learning on the job of automatic activities. To this aim new adaptive models based on anthologies and user stereotypes will be investigated. Furthermore, we are going to improve the processes automation through the integration, the coordination, and communication of both human and automatic task of business processes.

References

[1] AIMS: Adaptive Information System for Management of Learning Content. *Available at: http://wwwis.win.tue.nl:8080/AIMS, 2004*.

[2] L. Aversano, G. Canfora, A. De Lucia, P. Gallucci, "Business process reengineering and workflow automation: a technology transfer experience". *In The Journal of Systems and Software*, vol. 63 (1), 2002, pp.: 29-44.

[3] L. Aversano, A. De Lucia, M. Gaeta, P. Ritrovato, S. Stefanucci, M. L. Villani. "Managing Coordination and Cooperation in Distributed Software Processes: the GENESIS Environment". *In Software Process Improvement and Practice*, Vol. 9(4), Wiley, 2004, pp.:239-263.

[4] L. R. Bruce, and P. J. Sleeman. "Instructional Design: a primer". *In Greenwich, CT: Information Age Publishing*, 2000.

[5] F. Casati, G. Cugola. "Error handling in process support systems". *In Advances in Exception Handling Techniques*, A. Romanovsky, C. Dony, J.L. Knudsen, A. Tripathi (Eds), LNCS vol. 2022, Springer-Verlag, 2001.

[6] G. Costagliola, F. Ferrucci, G. Polese, G. Scanniello. "A Visual Language Based System for Designing and Presenting E-learning Courses". *In International Journal of Distance Education Technologies*, Vol. 3(1), 2005, pp.:1-19.

[7] G. Cugola, E. Di Nitto, A., Fuggetta. "The JEDI event-based infrastructure and its application to the development of the OPSS WfMS". *In IEEE Transactions on Software Engineering*, Vol. 27(9), 2001, pp.:827-850.

[8] A. De Lucia, R. Francese, G. Tortora. "Deriving workflow enactment rules from UML activity diagrams: a case study". *In Proceedings of Symposium on Human-Centric Computing Languages and Environments*, Auckland, New Zealand, 2003, pp.:211-218.

[9] E. Di Nitto, L. Lavazza, M. Schiavoni, E. Tracanella, M. Trombetta. "Deriving executable process description from UML". *In Proceedings of 24th International Conference on Software Engineering*, Orlando, Florida, ACM Press, 2002, pp.: 155-165.

[10] J. Eder, E. Panagos. "Towards Distributed Workflow Process Management". *AT&T Research Labs*, 1999.

[11] H. E. Eriksson, M. Penker. "Business modelling with UML". *Wiley Computing Publishing*, 2000.

[12] P.Heimann, G.Joeris, C. A. Krapp, B.Westfechtel. "DYNAMITE: dynamic task nets for software process management". *In Proceedings of the 18th International Conference on Software Engineering*, Berlin, Germany, 1996, 331-341.

[13] D. Jager, A. Schleicher, B. Westfechtel. "Using UML for software process modelling". *In Proceedings of ESEC/FSE'99*, Toulose, France, LNCS 1687, Springer, 1999.

[14] G. E. Kaiser, S. E.Dossick, W. Jiang, J. J. Yang, S. Xi Ye. "WWW-based Collaboration Environments with Distributed Tool Services". *In World Wide Web Journal*, Vol. 1(1), 1998, pp. 3-25.

[15] P. Loops, T. Allweyer. "Object orientation in business process modelling through applying event driven process chains (EPC) in UML". *In Proceedings of 2nd International Workshop on Enterprise Distributed Object Computing*, 1998.

[16] C. Marshall. "Enterprise Modellingwith UML: Designing Successful Software trough Business Analysis". Addison-Wesley, 2000.

[17] OMG, Unified Modelling Language Specification. Version 1.5, http://www.omg.org/technology/documents/formal/uml.htm, 2002.

[18] J. R. Thomson, and J. Cooke. "Generating Instructional Hypermedia with APHID". *In Hypertext 2000*. pp. 248-249.

[19] C. Vrasidas. "A Systematic Approach for Designing Hypermedia Environments for Teaching and Learning". *In International Journal of Instructional Media.* Available at http://www.cait.org/vrasidas/hypermedia.pdf, 2002.

[20] T. Winograd, F.Flores. "Understanding computers and cognition: a new foundation for design". Ablex publishing, Norwood, 1996.

[21] Workflow Management Coalition. "Workflow Management Coalition Interface 1: Process Definition Interchange Process Model". *Doc. no. WFMC-TC-1016-P*, 1999.

A Personalized E-Learning System Based on User Profile Constructed Using Information Fusion

Xin Li and Shi-Kuo Chang
Department of Computer Science,
University of Pittsburgh, USA,
{flying, chang}@cs.pitt.edu

Abstract

In this paper, we describe a personalized e-learning system which can automatically adapt to the interests and levels of learners. The system is designed based on the IEEE Learning Technology Systems Architecture (IEEE LTSA) to achieve high scalability and reusability. A feedback extractor with fusion capability is proposed to combine multiple feedback measures to infer user preferences. User profile, which stores user preferences and levels of expertise, is collected by user profiler to deliver personalized information using the collaborative filtering algorithm.

1. Introduction

Comparing with the traditional face-to-face style teaching and learning, e-learning is indeed a revolutionary way to provide education in life long term. Nowadays more and more people have benefited from various e-learning programs. However, high diversity of the learners on the Internet poses new challenges to the traditional "one-size-fit-all" learning model, in which a single set of learning resource is provided to all learners. In fact, the learners could have various interests; even sharing with the common interests, they may have different levels of expertise, and hence they can not be treated in a uniform way. It is of great importance to provide a personalized system which can automatically adapt to the interests and levels of learners.

User profiling is a promising approach towards the personalized e-learning systems where user profile including interests, levels and learning patterns can be assessed during the learning process. Based upon the profile, personalized learning resource could be generated to match the individual preferences and levels. Furthermore, learners with the common interests and levels can be grouped, and feedbacks of one person can serve as the guideline for information delivery to the other members within the same group.

In fact, user profiling is also the key process of many other applications; for example, the recommendation systems [1, 4, 8, 9] mainly depend on user profiles in terms of similarity and differences to provide particular suggestions. The personalized web search engine [11] can construct user profiles from browsing history and consequently provide personalized results to match the information needs of individuals. Comparing with these applications, user profiling is more feasible and important in e-learning system because learning is a much more continuous process than other activities such as online news reading and web searching.

Most approaches of user profiling are heavily depending on the user feedbacks to construct user profiles. The feedback can be assessed explicitly by rating, or implicitly by the user behaviors such as print and save. In this paper, we are not advocating either of these two approaches since both of them have significant advantage and disadvantages [11]. Instead we propose a system which can combine multiple feedback measures to get more complete and accurate profiles using the information fusion techniques.

Our e-learning system is designed based upon the IEEE Learning Technology Systems Architecture (LTSA), where multiple means of information delivery are provided including a chatting room, a customized web browser and whiteboard. A *feedback extractor* with fusion capability is designed to combine multiple feedback measures such as reading time, the number of scroll, print/save and relational index on chatting history. *User profile*, which stores user preferences and levels of expertise, is collected by *user profiler* to deliver personalized information using the collaborative filtering algorithm [8].

The rest of the paper is organized as follows:

Section 2 is for the related research. The system architecture based on the IEEE LTSA is described in Section 3, and then main components of our system are discussed in detail: *learning resources* and *user profile* in Section 4, *feedback extractor* in Section 5 and *user profiler* in Section 6. The experimental system and result analysis are described in Section 7, followed by the discussion and future research in Section 8.

2. Related Research

As the related research, the personalized e-learning system and recommendation system are discussed respectively. Information fusion, which is a relatively new concept in this domain, is also briefly reviewed.

2.1. Personalized E-Learning System

Blochl et al. [2] proposed an adaptive learning system which can incorporate psychological aspects of learning process into the user profile to deliver individualized learning resource. The user profile is placed in multi-dimensional space with three stages of the semantic decisions: cognitive style, skills and user type. However, both the means to acquire user's feedback and the algorithms to update user profile have not been addressed in the presentation.

SPERO [10] is a personalized e-learning system based on the IEEE Learning Technology Systems Architecture (LTSA). It could provide different contents for the foreign language learners according their interests and levels. The problem of SPERO system is that it is largely using questionnaires and e-surveys to build user profiles, which costs the users too much extra work.

2.2. Recommendation Systems

User profiling is the key process of recommendation systems, which collect user feedback for items in a given domain and assess user profiles in terms of similarities and differences to determine what to recommend. Depending on underline technique, recommendation systems can be divided into collaborative filtering-based [8] content-based [4] and hybrid [1, 9] approaches. Classified by means to acquire feedback, they can be categorized as explicit rating [1, 8, 9], implicit rating [8] and no rating needed [4] systems.

In fact, user's feedbacks are so important that only very few content-based recommendation systems require neither explicit rating nor implicit rating. For example, Surflen [4] is a recommendation system using data mining techniques to assess the association rules on web pages through user’s browsing history without the feedbacks. However, it's hard to find user's exact interests just based on the browsing history, since it always happens that users open a page they don't like or just by mistake. This problem becomes even more severe in the situation that the system is sparsely used.

GroupLens system [8], which filters Usenet news, is a collaborative filtering system using *n-nearest neighbor-based* algorithm. In this algorithm, user profile is assessed based on a subset of appropriate *n* users similar to this user. The early version of GroupLens gathers user's feedback only by explicit rate. However, observing the extra costs of the explicit rating, in the latest version it also uses reading time as an implicit indicator.

Fab system [1] is also using the collaborative filtering model, meanwhile introducing the content analysis by a “topic” filtering. Web pages are initially ranked by the topic filter and then sent to user's personal filters. Users are required to give an explicit rate, and this feedback is used to modify both the personal filter and the original topic filter.

2.3. Information Fusion

The key commonality underlying applications which require information fusion is that they need retrieve information on the same object from multiple data sources[5]. For example, in our approach, the multiple indicators are available to assess user preference; it is a fusion problem to combine them to get more complete and accurate results.

Information fusion is intensively investigated in sensor-based data processing systems such as intelligent surveillance systems, robotics vision and medical diagnoses systems, where multiple levels of fusion process are formulated and many algorithms have been developed[5, 6]. The problem we are tackling in this paper can be categorized as a decision-level identity fusion: the goal is a joint combined declaration from individual indicators. The techniques used on this level include voting, Bayesian inference, Dempster Shafer's method, and so on.

3. System Architecture

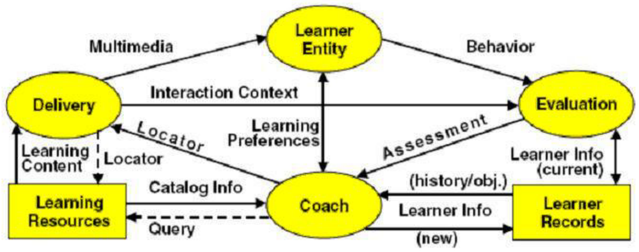


Figure 1 The IEEE Learning Technology Systems Architecture (IEEE LTSA)

The IEEE Learning Technology Systems Architecture (IEEE LTSA) [7] is a component-based framework for general learning system with high scalability and reusability. It includes three types of components shown in Figure 1:

- *Processes*: learner entity, evaluation, coach and delivery;
- *Stores*: learning records and learning resources;
- *Flows*: learning preference, behavior, assessment information, learner information, query catalog info, locator, learning content, multimedia and interaction context.

where the *processes* and *stores* exchange information through the *flows*.

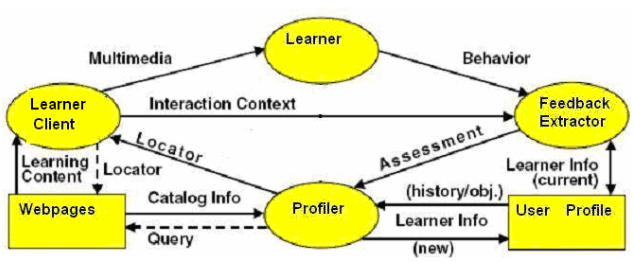


Figure 2 The Architecture of Our System

In our system, we instantiate the abstract conceptual models in IEEE LTSA by the real components shown in Figure 2:

- *Learning Resources* are represented as *Webpages* including multimedia resources such as video and audio clips.
- *Learner's Record* is implemented by *User Profile*, which stores performance, preference, etc.
- *Evaluation Entity* is implemented by *Feedback Extractor*, which can infer learner preference by fusing the multiple feedback measures.
- *Delivery Entity* is implemented by *Learner Client*.
- *Coach Entity* is implemented by *User Profiler* which interacts with other components:
  1. Receiving the preference assessment from *Feedback Extractor*.
  2. Assessing/Updating *User Profile*.
  3. Providing a guideline for information delivery for *Learner Client*.

The link between *Coach* and *Learner Entity* is removed since the feedbacks are collected implicitly.

4. Learning Resources and User Profile

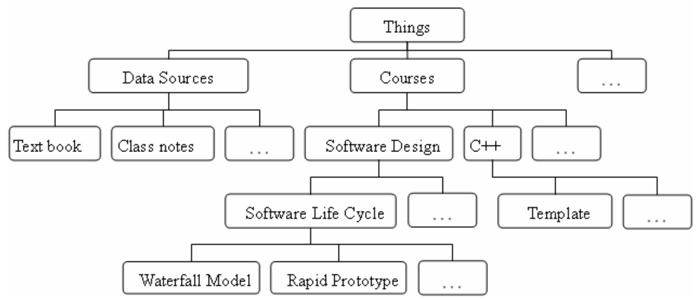


Figure 3 An Example of Ontology Knowledge Base

Learning Resources (e.g. webpages) are organized by the topics which are structured in ontology knowledge base (OKB). An example of OKB is shown in 3. The each topic is attached with several keywords.

Definition 4.1 a webpage *pg* is a 4-tuple:

$$pg = <id, co, tp, l>$$

where

- *id* is a unique identification number.
- *co* is the content of *pg*.
- *tp* is the topic.
- *l* is the expertise level of *pg* such as beginning, intermediate and advanced.

The user profile is defined as follows:

Definition 4.2 A user profile *upf* is a 4-tuple:

$$upf = <id, bh, ch, ls>$$

where

- *id* is a unique identification number.
- *bh* is the browsing history represented as $\{<pg_1,r_1>, <pg_2,r_2>, \ldots, <pg_n, r_n>\}$, where pg_i is a webpage read by the user; r_i is user preference on pg_i in [0..1] (i=1, 2, …, n).
- *ch* is the chatting history in natural languages.
- *ls* is the levels of expertise represented as $\{<tp_1,l_1>,$

<tp_2,l_2>, …, <tp_n, l_n>}, where tp_i is a topics; l_i is the levels on tp_i in terms of beginning, intermediate and advanced (i=1, 2, …, n).

5. Feedback Extractor

Feedback Extractor collects feedbacks to make a final assessment of user preference. In this section, firstly the feedback measures are described. Next, a fusion model is proposed to fuse these measures.

5.1. Feedback Indicators

Definition 5.1 Feedback indicator for a webpage *pg* is a function which returns 0 or 1, where 0/1 means the negative/positive correlation with user preference.

Four implicit feedback indicators are employed in our system including: reading time, scroll, print/save and relational index.

- Reading Time: return 1 if user read *pg* longer than φ_t, where φ_t is a predefined threshold; 0 otherwise.
- Scroll: return 1 if the number of user scrolls (either mouse or keyboard Pageup/Pagedown) on *pg* is greater than φ_s, where φ_s is a predefined threshold; 0 otherwise.
- Print/Save: return 1 if user prints/saves *pg*; 0 otherwise.
- Relational Index: return 1 if keywords of *pg* appear in user's chatting history *ch* more than φ_r times, where φ_r is a predefined threshold; 0 otherwise.

Although these indicators could be pretty much context-related, we can make them objective measures by normalizing the learning resources. For example, the webpages are trimmed to have roughly the same length and layout, so reading time could effectively indicates the interest of users.

5.2. Fusion Model

To compile all these feedbacks for a final assessment of user preference, we map the problem into an information fusion process. Define ***H*** is a hypothesis that user has positive preference, given independent indicators $I_1, I_2, ..., I_n$ (n>1), the posteriori probability of P(**H**| $I_1, I_2, ..., I_n$) is the joint declaration, which can be assessed using Bayesian method:

$$P(H \mid I_1, I_2, ..., I_n) = \frac{P(I_1, I_2, ..., I_n \mid H) \cdot P(H)}{P(I_1, I_2, ..., I_n)}$$

$$= \frac{\prod_{i=1}^{n} P(I_i \mid H) \cdot P(H)}{\sum_{i=1}^{n} (P(I_i \mid H) \cdot P(H) + P(I_i \mid \neg H) P(\neg H))}$$

where

- $P(\neg H) = 1 - P(H)$
- $P(I_i \mid H)$ and $P(I_i \mid \neg H)$ are probability of the observing indicator I_i given *H*.
- $P(H)$ is the priori probability of the hypothesis *H* (without having observed the evidence)

$P(I_i \mid H)$, $P(I_i \mid \neg H)$ and $P(H)$ are called *model parameters* which can be assessed through the statistical analysis on training data.

6. User Profiler

User Profiler is a core component in our system. In this section, we firstly present a brief overview on it in terms of the input and output. Next, the two main functions are discussed in detail.

6.1. Overview

Briefly *User Profiler* has two tasks:

1. Assessment of Expertise Level
   - Input: user's browsing history with the preference assessed by *Feedback Extractor*.
   - Output: user's levels of expertise.
2. Providing Guideline for Delivery
   - Input: user's browsing history and levels of expertise.
   - Output: a list of webpages, which are potentially interesting to users.

6.2. Assessment of Expertise Level

Basically expertise levels are determined by the average preferences. The webpages user has read on any topic *tp* could have different levels in terms of beginning, intermediate and advanced. The level of the user on *tp* is the one which has the highest average preference.

6.3. Guideline for Delivery

The information delivery is based on the collaborative filtering algorithm [8]. Given two users U_1 and U_2, $pg_1, pg_2, …, pg_n$ are the common pages they both read, with the feedback $x_1, x_2, \ldots, x_n$ and $y_1, y_2, \ldots, y_n$ respectively. Assume the average feedbacks of the page $pg_1, pg_2, \ldots, pg_n$ are $\varpi_1, \varpi_2, \ldots, \varpi_n$, a similarity function S on U_1 and U_2 is defined using Pearson correlation coefficient:

$$S(U_1, U_2) = \frac{\sum_{i=1}^{n}(x_i - \varpi_i) \times (y_i - \varpi_i)}{\sqrt{\sum_{i=1}^{n}(x_i - \varpi_i)^2 \times \sum_{i=1}^{n}(y_i - \varpi_i)^2}} \quad (1)$$

Given any active user U_x, using (1) could find the n users who have the highest similarity, named n neighbors $\{U_1, U_2, \ldots, U_n\}$ of U_x, the preference of U_x on page pg — p_x can be predicted by the preferences of the neighbors which is already known, denoted as $p_1, p_2, \ldots, p_n$. Given ϖ is the average rating on page pg,

$$p_x = \varpi + \frac{\sum_{i=1}^{n} p_i \times S(U_x, U_i)}{\sum_{i=1}^{n} S(U_x, U_i)} \quad (2)$$

The webpages with the highest interest predictions are the potential interesting pages for U_x, which will be delivered without requests.

7. Prototype System and Experiments

In this section, we briefly introduce our prototype system, followed by the experiments and some preliminary results.

7.1. Prototype System

A prototype system has been implemented. The system diagram is shown in Figure 4. Figure 5 shows the main interface of learner client. Several means are provided for the information delivery including a chatting room (right bottom), a customized web browser (left) and whiteboard (right top). With the help of communication server, multiple feedback measures are recorded for feedback extractor; user profiler can push the web pages without requests.

7.2. Experiments

For system training purpose, we ask a group of students to do the following experiments:

- Step 1: select a topic such as "E-R diagram" and "C++", let the students indicate their levels on it in terms of beginning, intermediate, or advanced.
- Step 2: ask the students to use leaner client (Figure 5) reading the prepared webpages.
- Step 3: require the students seriously rating the interest on every article they have read from 1 (the least) and 5 (the most).

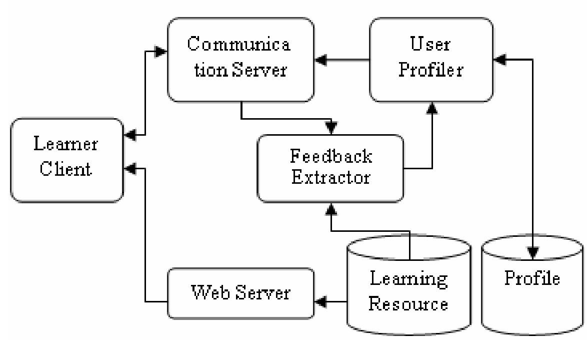


Figure 4 System Diagram

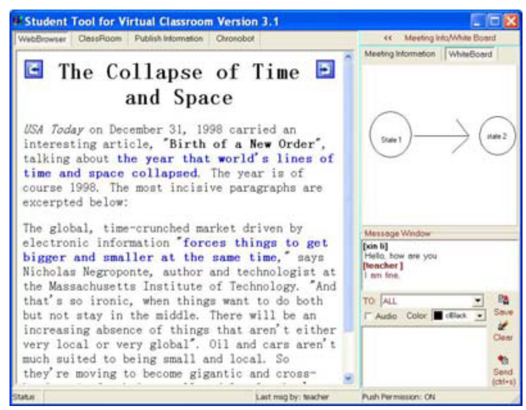


Figure 5 Learner Client

7.3. Preliminary Results and Analysis

Some preliminary results of the experiments have been collected. Figure 6 shows three indicators with explicit rating (the relational index is not employed in the experiments). The thresholds of the reading time φ_t and scroll φ_s are simply determined by the average value of medium evaluation (rating 3). The *model parameters* of the fusion model discussed in Section 5.2 can be assessed based upon these thresholds.

Compared with the provided levels by users, the level assessment algorithm described in Section 6.2 has 83.2% accuracy. However, the evaluation means of information delivery are still under development.

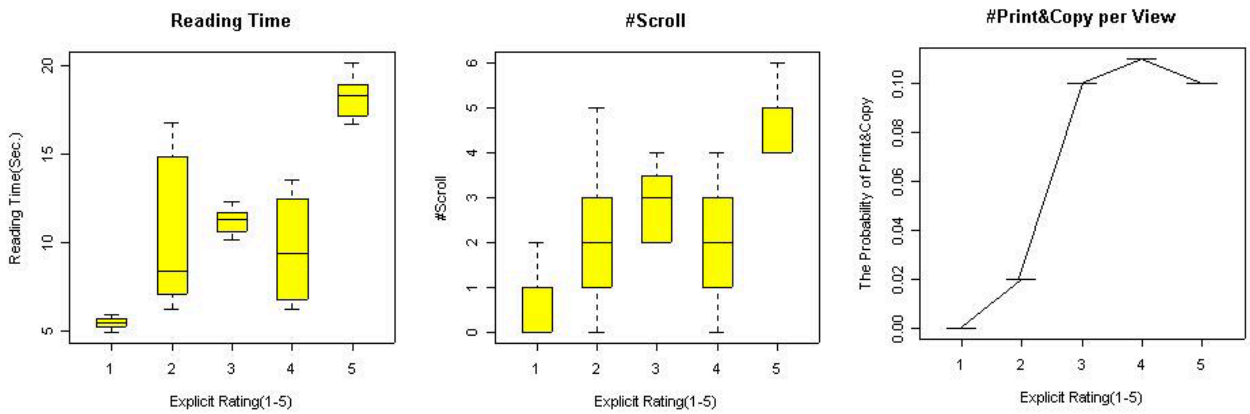


Figure 6 Feedback Measures vs. Explicit Rating

8. Discussion and Future Research

In this paper, we described our current ongoing research on the personalized e-learning system: the system is designed based upon the IEEE LTSA architecture; a *feedback extractor* with fusion capability is introduced to combine multiple feedback measures; user profile is collected by *user profiler* to deliver personalized information; the prototype system and preliminary results are presented.

However, the fourth indicator – relational index is not testified in the experiments yet. Furthermore, the usability of the system has not been fully verified by the end users, especially for the quality of information delivery. On the other hand, although ontology knowledge is used for the content classification, the structure of it could be much more complicated and the usage of it can be extended to the feedback extracting and user profiling. All these could lead to some very interesting topics and will be the subjects of our future research.

9. Acknowledgement

This research is a part of Chronobot project [3], which is supported in part by the Industry Technology Research Institute (ITRI) and the Institute for Information Industry (III) of Taiwan. We would like to thank Chieh-Chih Chang and Jui-Hsin Huang for the valuable discussion and comments.

References

[1] M. Balabanovi and Y. Shoham. Fab: content-based, collaborative recommendation. *Communication of the ACM*, 40(3):66–72, 1997.

[2] M. Blochl, H. Rumershofer, and W. Wob. Individualized e-learning systems enabled by a semantically determined adaptation of learning fragments. In *Proceeding of the 14th international workshop on Database and Expert Systems Applications*, pages 640–645, 2003.

[3] S. K. Chang. A chronobot for time and knowlege exchange. In *Tenth International Conference on Distributed Multimedia Systems*, pages 3–6, San Francisco Bay, CA, Sep. 2004.

[4] X. Fu, J. Budzik, and K. J. Hammond. Mining navigation history for recommendation. In *Proceedings of the 5th international conference on Intelligent user interfaces*, pages 106–112, 2000.

[5] D. L. Hall and J. Llinas, editors. *Handbook of Multisensor Data Fusion*. CRC Press, New York, 2001.

[6] D. L. Hall and S. A. H. McMullen, editors. *Mathematical Techniques in Multisensor Data Fusion*. Artech House Publishers, 2004.

[7] IEEE LTSC P1484.1. http://ltsc.ieee.org/wg1/index.html.

[8] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl. Grouplens: applying collaborative filtering to usenet news. *Communication of the ACM*, 40(3):77–87, 1997.

[9] S. E. Middleton, N. R. Shadbolt, and D. C. D. Roure. Ontological user profiling in recommender systems. *ACM Trans. Inf. Syst.*, 22(1):54–88, 2004.

[10] SPERO: Tele-Informatics System for Continuous Collection, Processing, Diffusion of Material for Teacher Training in Special Education. http://www.image.ntua.gr/spero.

[11] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *Proceedings of the 13th international conference on World Wide Web*, pages 675–684, 2004.

An Application of XML on Network Data Model for Data Conversion

J. K. Chen*
Department of Information Management
Chaoyang University of Technology
jkchen@mail.cyut.edu.tw

C. J. Liu
Department of Information Management
Chaoyang University of Technology
s9214623@mail.cyut.edu.tw

Abstract

XML has become a standard medium for data exchange in various business applications. The one-to-many relationship of data can be represented conveniently by XML but the many-to-many relationship of data is hard to be described by XML. Most researches focus on data conversion between XML and relational data model. Network data model is an alternative for saving data with many-to-many relationships and is still being used in several enterprises or organizations. A mechanism for data conversion between XML and network data model has been proposed. However, the problem of the loss of data type for some data is not solved yet. In this paper, we propose a method to preserve the original data types for each data item during data conversion.

Keywords: XML, XML Schema, Network Data Model, DBD, Data Conversion, Algorithm

1. Introduction

Today, one of the most important business transactions is data exchange between enterprises or organizations. Two companies can quickly send business data to each other to finish a trade via Internet. The typical mediums for business data exchange are text file, PDF, HTML, SGML, XML, XHTML, [8], [13], [14], etc. while XML [19] is the most popular one [1], [4], [12], [15], [17]. The relationship among data in database has two kinds, one-to-many and many-to-many. XML is good at carrying data with one-to-many relationship because of its hierarchical structure but is hard to describe data with many-to-many relationship. Several methods [5], [9], [10] solve this problem of converting data with many-to-many relationships between XML documents and databases based on the relational data model. However, some enterprises or organizations now still use the database based on the network data model (**NDM**) [11] derived from the hierarchical data model (**HDM**). This is because IBM improved its IMS [6], [7], a famous platform for HDM-based databases, to IMS/2 to contain data with many-to-many relationships.

With the mechanism of [2], enterprises using NDM-based databases now can exchange data by XML documents. However, a new problem occurs about the loss of data type. Although data values in database can be converted correctly into an XML document, the data types of data values may be misinterpreted because all data values are presented by character string in an XML document. Except character string, there are several common used data types such as integer, float, Boolean, date, etc. An error may occur if a data item with data type of non-character string is interpreted as character string. Although [3] proposed a method to hold the original data type for data exchange between HDM-based databases and XML documents, it cannot be applied to data exchange between NDM-based databases and XML documents because data in HDM-based databases has no many-to-many relationships. In this paper, we use DBD and XML Schema, two auxiliary files for NDM-based database and XML document, respectively, to preserve the original data type of each data item. In fact, there is another alternative auxiliary file, DTD, for NDM-based database. We choose XML Schema because it supports more than 44 data types than DTD does. Two new algorithms are proposed for the conversion between DBD and XML Schema. By the algorithms in [2] and the new ones proposed in this paper, enterprises can exchange data with many-to-many relationships correctly without worrying the problem of the loss of data type.

2. Previous Work

2.1. Network Data Model and DBD

Derived from HDM, NMD basically has the same characteristics with HDM except the extra feature of saving the many-to-many relationship of data. No representative production, the NDM-based database is always implemented at HDM-based database platform such as IMS/2. In 1970, IBM improved its IMS to IMS/2 by adding *the logical relationship mechanism* [11] to make IMS to be able to hold the many-to-many relationship of data. In NDM, each individual entity type is implemented as a *segment* [6] that contains several segment instances. The many-to-many relationship of data is established by using the logical relationship mechanism associated with five built-in pointers in each segment instance [2]. An example of an NDM-based database is shown in Figure 1. The root segment "department" has two child segments "employee" and "project." As a bridge, the segment "participation" establishes the many-to-many relationship between segments "employee" and "project."

To describe data, the Database Task Group of CODASYL specified the Schema Data Description Language (DDL) for the data structure of database. The DataBase Description block (**DBD**) [6] is the DDL of NDM-based databases. A DBD, an auxiliary file to a corresponding database, can describe a database about the structure of the database, the schema of each segment, and the data type of each field in each segment. The structure of a DBD is described as followed. First the DBD name is defined then some segments followed by more than one field are defined sequentially. The

DBD name is the same as the database name. Defined by the keyword "SEGM," a segment includes the segment name (the "NAME"), the total byte number of the fields that constitute the segment (the "BYTES"), and the parent segment name of the segment (the "PARENT"). Defined by the keyword "FIELD," each field has three items: the field name (the "NAME"), the data type (the "TYPE"), and the data length (the "BYTES"). The associated DBD of the NDM-based database in Figure 1 is shown in Figure 2. The DBD is declared as "departmentdb" containing the root segment "department." The segment "employee" has three fields that occupy 54 bytes and its parent is segment "department." The three fields of the segment "employee" are "eID," "ename," and "salary" where the length of field "salary" is 12 bytes and the data type of field "salary" is float. Note that the segment "participation" has two parents. The first one is its physical parent while the second one is its logical parent [2]. The keyword "LCHILD" for segment "project" is used to describe the logical child, segment "participation," of segment "project."

2.2. XML and XML Schema

Derived from SGML, XML is proposed by W3C in 1998. XML has become a standard media because of its self-description in both data and structure. The detailed descriptions of XML can be found in [18], [19]. A well-formed XML document is composed of the root element and its descendant elements. These elements constitute a hierarchical structure to represent the one-to-many relationship of a parent element and its child elements. Published by W3C, XML Schema [20], [21], [22] can describe an XML document about the structure of the document, the schema of each element, and the data type of each attribute in each element. An XML Schema is an auxiliary file to an XML document just like a DBD to an NDM-based database. The characteristics of XML Schema are described as follows. XML Schema supports more than 44 data types including integer, string, decimal, date, etc [22]. The XML Schema namespace can prevent the ambiguity that the same element or attribute names in an XML document have different meanings.

3. Conversion Technique

To make the data in an NDM-based database to be interpreted correctly, the associated DBD file of the database must be exactly converted into an XML Schema file to assist with the XML document obtained from the NDM-based database. Likewise, the auxiliary file, XML Schema, also need to be converted exactly into the original DBD file when the data in the XML document is restored to an NDM-based database. Therefore, we propose two algorithms, DBDToXSchema and XSchemaToDBD, to perform the above two objectives, respectively. The two algorithms are described as follows.

3.1. The DBDToXSchema algorithm

The algorithm DBDToXSchema is used to convert an input DBD file into an output XML Schema file. This algorithm is divided into two phases. The first phase translates the contents of the DBD file into a binary tree. The second phase translates the binary tree into the desired XML Schema file. The binary tree is used to save the structure of the segment data in the DBD. Beside the general attributes data, some meta attributes such as "LP_path," "PP_path," "LTF_path," and "LCF" are added to the XML Schema file in order to keep the many-to-many relationship of segments in the DBD file. The detailed contents of algorithm DBDToXSchema are listed as follows.

```
Algorithm DBDToXSchema(D: DBD_FILE, X:
XML_SCHEMA)
Input: D,    Output: X
begin
  T: BINARY_TREE; ptr: BINARY_TREE;
  line_string: STRING;
  // Phase 1: convert the DBD to a binary tree //
  create a node for T;
  while D is not empty, do
    read a line data from D to line_string;
    if line_string is the DBD name, then
      put the DBD name to the data item of T;
      create a node as the left child of T;
      ptr points to the left child node;
    end if;
    if line_string is a segment declaration, then
     if the value of keyword "PARENT" in line_string
     is not '0', then
        traverse T to find the node n whose name value
        of the data item is the same as the value of
        'PARENT" in line_string;
        if the left pointer of n is nil, then
          create a node as the left child of n;
          ptr points to the left child node of n;
        else ptr points to the left child node of n and
            continues pointing to the right child node of
            ptr until ptr's right pointer is nil;
             create a node as the right child of ptr;
             ptr points to its right child node;
        end if;
      end if;
      put the data of the current segment name, total
      bytes, and parent segment to the data item of ptr;
      read the field definitions of the segment from D
      and put them to a list L;
      put L to the field list item of ptr;
      if the keyword "PARENT" has two values in
      line_string, then
        add the meta fields, "PP_path", "LP_path", and
        "LTF_path", of the current segment to the field
        list item of ptr;
      end if;
    end if;
    if line_string is the keyword "LCHILD" declaration
      add the meta field "LCF_" + the logical child name
      of the current segment to the field list item of ptr;
    end if;
  end while;
  // Phase 2: convert the binary tree into an XML
  Schema file //
  write the required information of schema tag to X;
```

```
  traverse all nodes of T with the pre-order sequence,
  for each node n, do
    read the value item of n and write to X as an
    element definition;
    read the field list item of n and write to X as
    attribute definitions;
  end for;
end DBDToXShema.
```

3.2. The XSchemaToDBD module

The algorithm XSchemaToDBD is used to convert an input XML Schema file into an output DBD file. This algorithm is divided into two phases. The first phase translates the contents of the XML Schema file into an m-way tree. The second phase translates the m-way tree into the desired DBD file. The detailed contents of algorithm XSchemaToDBD are listed as follows.

```
Algorithm XSchemaToDBD(X: XML_SCHEMA, D:
DBD_FILE)
Input: X, Output: D
begin
  T: M-WAY_TREE; ptr: M-WAY_TREE;
  tag_string: STRING; line_string: STRING;
// Phase 1: Convert the XML Schema to an m-way tree //
  read the start tag of the root element from X and write
  to D as the DBD name;
  create a node for T; ptr points to T;
  while X is not empty, do
    read a tag from X to tag_string;
    if tag_string is a start tag, then
      if tag_string is an element, then
        create a node as a child of ptr and then set ptr
        point to that node;
        put the element name to the data item of ptr;
        if the element has a parent element, then
          set the parent element name of the element to
          the data item of ptr;
        else set '0' to the data item of ptr;
        end if;
      end if;
      if tag_string is an attribute, then
        create a link list unit u and put the attribute
        name to u; add u to the field list item of ptr;
      end if;
      if tag_string is the data type declaration of the
      attribute, then
        compute the byte length of the data type;
        put the byte length and data type to the link list
        unit that contains the attribute name in the field
        list item of ptr;
      end if;
      if tag_string is the data length declaration of the
      attribute, then
        recompute the byte length of the attribute;
        update the byte length of the link list unit that
        contains the attribute name in the field list item
        of ptr;
      end if;
    else //tag_string is a close tag//
      if tag_string is an element close tag "</element>",
      then
        ptr points to the parent node of ptr;
      end if;
    end if;
  end while;
  // Phase 2: write the m-way tree T to a DBD file //
  traverse all the nodes of T with the depth-first
  sequence,
  for each node n, do
    if there is no meta attributes in n, then
      write the data item of n to D as a segment definition;
      write the field list item of n to D as field
      definitions;
    else // a logical parent or a bridge segment //
      if the meta attribute "LCF_" + a logical child
      name is found in n, then
        write the data item of n to D as a segment definition;
        write the field list item of n to D as field definitions;
        set the keyword "LCHILD" and the logical
        child name of n to line_string;
        write line_string to D;
      end if;
      if the meta attribute "PP_path" is found in n, then
        if n's parent has no the meta attribute "LCF_" +
        n's name, then
          search all nodes of T to find a node N in
          which the field list item has the meta
          attribute: "LCF_" + n's name;
          add N's name to the data item of n as the
          second value of the keyword "PARENT";
          write the data item of n to D as a segments definition;
          write the field list item of n to D as field definitions;
        end if;
      end if;
    end if;
  end for;
end XShemaToDBD.
```

4. Conclusion

The enterprises using NDM-based databases can exchange data each other for business transaction by the algorithms in [7] to convert data of NDM-based database and XML document. However, a serious loss may occur if business data in an XML document is interpreted without considering the frequent used data type such as integer, float, date, etc. It is very important to save the data types when data is translated from the source database into an XML document and to recover the data types when data is restored from an XML document to the destination database. In this paper, we propose two algorithms for data conversion between DBD and XML Schema files to assist the data exchange of NDM-based database and XML document. In the future work, we should implement the proposed algorithms and design several experiments to prove the algorithms to be workable.

REFERENCE

[1] E. Bertino and B. Catania, "Integrating XML and databases," *IEEE Internet Computing,* Vol. 5, Issue: 4, 2001.

[2] J. K. Chen and C. J. Liu, "A Mechanism for Data Conversion between XML and Network Data Model," *proceedings of the 2004 International*

Computer Symposium, Dec. 15-17, Taipei, Taiwan, ROC, Session TA-G1, 2004

[3] J. K. Chen and M. J. Liu, "An Application of XML on Hierarchical Database for Data Exchange," proceedings of the 2003 National Computer Symposium, Dec. 18-19, Taichung, Taiwan, ROC, Session B3, OT-058, 2003

[4] F. M. Cleveland, "Information exchange modeling (IEM) and extensible markup language (XML) technologies," *IEEE Power Engineering Society Winter Meeting,* Vol. 1, pp. 592-595, 2002.

[5] J. Fong, F. Pang, and C. Bloor, "Converting relational database into XML document," *Proceedings of 12th International workshop on Database and Expert Systems Applications,* 2001.

[6] IBM, *IMS Primer,* http://www.redbooks.ibm.com

[7] IBM, *IMS/ESA V5 Admin Guide: DB,* http://www-306.ibm.com/software/data/ims/v5pdf/DFSA10C6.PDF

[8] G. Kappel, S. Rausch-Schott, S. Reich, and W. Retschitzegger, "Hypermedia document and workflow management based on active object-oriented databases," *Proceedings of the Thirtieth Hwaii International Conference, System Sciences,* Vol. 4, pp. 377-386, 1997.

[9] J. S. Kim, W. Y. Lee, and K. H. Lee, "The cost model for XML documents in relational database systems," *ACS/IEEE International Conference, Computer Systems and Applications,* pp. 185-187, 2001.

[10] T. Kudrass, "Management of XML documents without schema in relational database systems," *Information and Software Technology,* Vol. 44, Issue: 4, pp. 269-275, 2002.

[11] T. William Olle, *The Codasyl Approach to Data Base Management*, John Wiley & Sons, 1978.

[12] J. Singh, "XML for power market data exchange," *IEEE Power Engineering Society Winter Meeting,* Vol. 2, pp. 755-756, 2001.

[13] M. B. Spring, "Reference model for data interchange standards," *Computer,* Vol. 29, Issue: 8, pp. 87-88, 1996.

[14] R. Summers, J. J. L. Chelsom, D. R. Nurse, and J. D. S. Kay, "Document management: an Intranet approach," *IEEE the 18th Annual International Conference, Engineering in Medicine and Biology Society, Bridging Disciplines for Biomedicine,* Vol. 3, pp.1236-1237, 1997.

[15] M. Sundaram and S. S. Y. Shim, "Infrastructure for B2B exchanges with RosettaNet," *The third International Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems,* pp. 110-119, 2001.

[16] R. W. Taylor and R. L. Frank,"CODASYL Data-Base Management System," ACM Computing Surveys, Vol. 8, Issue: 1, pp. 67-103, 1976

[17] S. Wegener and D. Davis, "XML TPS data exchange," *IEEE AUTOTESTCON proceedings of Systems Readiness Technology Conference,* pp. 605-615, 2001.

[18] W3C, Extensible Markup Language (XML) 1.0 W3C Recommendation, 10 February 1998 *http://www.w3.org/TR/1998/REC-xml-19980210,* 1998.

[19] W3C, Extensible Markup Language (XML) 1.1W3C Recommendation, 04 February 2004, *http://www.w3.org/TR/2004/REC-xml11-20040204/*, 2004

[20] W3C, XML Schema Part 0: Primer W3C Recommendation, 2 May 2001, *http://www.w3.org/TR/2001/REC-xmlschema-0-20010502*, 2001.

[21] W3C, XML Schema Part 1: Structures W3C Recommendation, 2 May 2001, *http://www.w3.org/TR/2001/REC-xmlschema-1-20010502*, 2001.

[22] W3C, XML Schema Part 2: Datatypes W3C Recommendation, 2 May 2001, *http://www.w3.org/TR/2001/REC-xmlschema-2-20010502*, 2001.

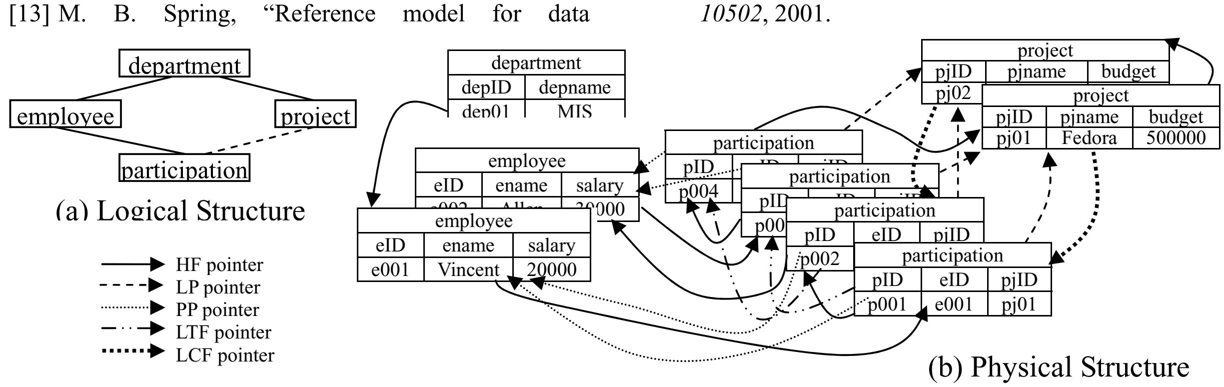


Figure 1 Logical Structure and Physical Structure

```
DBD
NAME=departmentdb
SEGM
NAME=department, BYTES=42, PARENT=0
FIELD NAME=depID, BYTES=12, TYPE=C
FIELD NAME=depname, BYTES=30, TYPE=C
SEGM
NAME=employee, BYTES=54,
  PARENT=department
FIELD NAME=eID, BYTES=12, TYPE=C
FIELD NAME=ename, BYTES=30, TYPE=C
FIELD NAME=salary, BYTES=12, TYPE=F
```

```
SEGM
NAME=participation, BYTES=36,
  PARENT=((employee),(project, departmentdb))
FIELD NAME=pID, BYTES=12, TYPE=C
FIELD NAME=eID, BYTES=12, TYPE=C
FIELD NAME=pjID, BYTES=12, TYPE=C
SEGM
NAME=project, BYTES=52, PARENT=department
FIELD NAME=pjID, BYTES=12, TYPE=C
FIELD NAME=pjname, BYTES=30, TYPE=C
FIELD NAME=budget, BYTES=10, TYPE=F
LCHILD=(participation, departmentdb)
```

Figure 2. A DBD File.

Protection of Virtual Property in Online Gaming

Ronggong Song, Larry Korba, George Yee, and Ying-Chieh Chen

Institute for Information Technology
National Research Council of Canada
Ottawa, Ontario K1A 0R6, Canada
{Ronggong.Song, Larry.Korba, George.Yee, Bomy.Chen}@nrc.ca

***Abstract*—Along with the success of the massively multiplayer role-playing gaming industry in Asia, online gaming-related crimes have grown at an amazing rate. Most of the criminal cases are related to virtual properties since markets have developed for the virtual properties giving them real world values. There has been little research and resulting technologies for MMRPG virtual property protection. In order to reduce the crimes and protect online gaming systems, one potential solution is protecting the virtual properties in online gaming systems. In this paper, we propose a virtual property management language to meter the use of virtual property. The language provides a framework for managing the use of virtual properties and recording the history of transactions to trace the life of virtual properties.**

1 Introduction

The Massively Multiplayer Online Role-Playing Gaming (MMORPG) is a popular computer entertainment in Asia. Played by more than one person over the Internet, MMORPG has turned into a successful business. According to the forecast of DataMonitor.com, the global online gaming market will grow to $3.2 billion and 113 million players in 2005 [1]. However, many problems have arisen and greatly influenced our society with the growth of online gaming. For instance, based on the annual criminal statistics report of Taiwan [2], over 30% criminal cases are related to online gaming. In addition, our further research [3, 4] has shown that most of the online gaming crimes are related to misappropriation of virtual properties since some MMORPG virtual properties have very high monetary values for trading in the marketplace. According to the estimation of NewScientist.com, real-world sales of virtual resources gained within MMORPGs has surpassed $100 million worldwide [5]. Another reason is that the systems themselves only have weak protection for virtual properties. Until now there has been little published research or technologies proposed for virtual property protection.

In order to manage and protect the virtual properties in the MMORPGs, we propose a virtual property management language (VPML) in this paper. By investigating the management requirements of virtual properties in MMORPGs, we define the core models for VPML with provisions for security mechanisms. Key entities include: trading agreement entity, history event entity, and security entity. VPML could provide different levels of security protection for virtual properties by choosing different security mechanisms in order to get a balance among required security level, efficiency, and scalability. This makes the systems more flexible, for instance, we could choose a higher level of security protection for the valuable virtual properties, and a lower level of security protection for those virtual properties of less value.

The rest of this paper is organized as follows. Some background research on MMORPG is briefly described in Section 2. In Section 3, the management requirements for virtual property are analyzed and summarized. In Section 4, the Virtual Privacy Management Language is described including VPML models and the requirements of the MMORPG systems. Several VPML XML examples are introduced in Section 5 to demonstrate how VPML works under the different management scenarios. The conclusions are presented in Section 6.

2 MMORPG

MMORPG is one kind of online gaming played over the Internet by many players from geographically diverse locations. The architecture for MMORPGs is based on the client-server technology. Figure 1 depicts the MMORPG network architecture. Currently, most MMORPGs are commercial in which the players must pay a network connection fee for operating their virtual characters and accumulating related virtual properties.

Since the availability of precious virtual properties is limited and some of it requires expenditure of considerable time and energy to develop, players who desire those properties have resorted to trading with players who have already gained desirable virtual property. In fact, the growth of virtual property markets has swollen the dollar value of desirable properties. For instance, a virtual “dragon knife” and a “royally invincible claw” recently received bids of $US 4,800 and $US 4,270 respectively at an online auction [6]. Trading of virtual property has become an extension to entertainment experience and common practice with MMORPGs, commencing with games such as Sony’s EverQuest [7]. For players, using cash to purchase virtual property can save time and vigor, immediately upgrading their virtual skills or degrees. Some players even run businesses assisting other players in reaching certain higher levels of the game. In Lineage II for example, reaching level 50 requires a fee of $945 within 15 to 20 days [8]. The online gaming industry has evaluated various trading models and methods. It is unlikely that game vendors will revise the rules in order to eliminate trading or exchanging. Players who want to trade virtual property can utilize diverse trading channels to sell, purchase, or exchange. For solving these trading dilemma, ItemBay.com [9] was built to provide a trading platform for players who want to exchange, sell, or purchase virtual properties in Korea and Taiwan since 2001. With this platform, more and more players can earn profits from MMORPGs at a lower risk than previous methods. When virtual properties become valuable to players and attain high real-world values, MMORPG becomes no longer just the entertainment. Within the marketplace for virtual properties, where there is the involvement of real money, conflicts and criminal behaviors can arise.

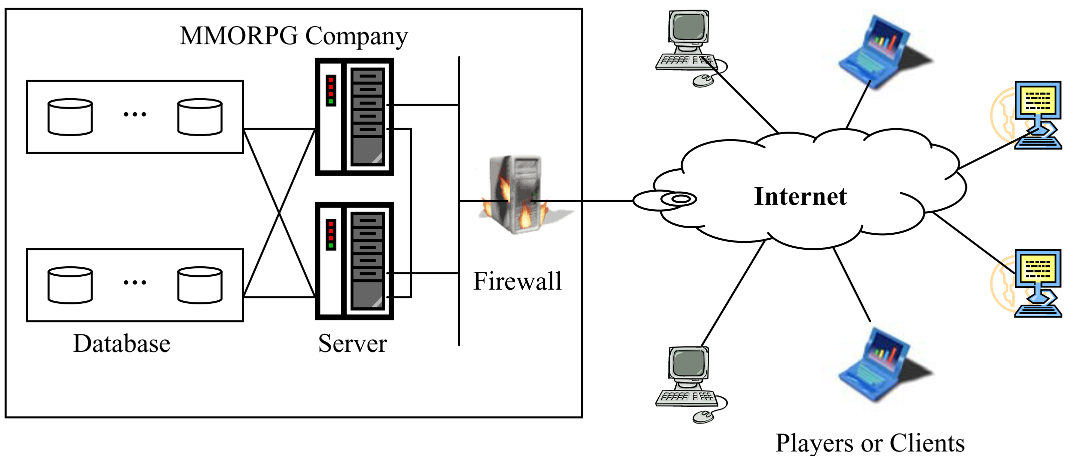


Fig. 1. MMORPG Network Architecture

3 Security Characteristics of Virtual Property

3.1 Security Analysis

In existing MMORPGs, most systems usually use a log file to offer some prospect of virtual property protection. For instance, recording the user's login information and activities provides some security protection for the user's account such that the virtual properties under the account get appropriate protection indirectly. Unfortunately this weak protection creates problems in the protection of virtual properties, especially in cases of fraud and stolen identity [3, 4]. In addition, existing systems have the following disadvantages for the virtual property protection.

- **Hard Tracing:** Since the log file contains a large variety of log messages, it makes the system harder to trace some special events related to the virtual property. The system may require data mining technologies for this purpose. This would often happen when the investigators or court officials require some evidence in the prosecution of online gaming criminal cases. In such cases, many of the other log messages would be useless for this purpose. How to get terse electronics evidence accurately and quickly is a big problem with current MMORPGs.
- **Weak Security Protection:** Current systems do not provide special security protection technologies for virtual properties. Unfortunately, because of this, the log file cannot provide strong evidence for the online gaming criminal cases. For instance, if a player steals another player's virtual property, s/he can claim that s/he has paid for it since the log file usually does not record the trading process, especially the trading agreement. The trading agreement may require some special security functions like a digital signature. Considering privacy issues, some players may want to hide some sensitive information related to the virtual property from other players. Moreover, the log file itself may be attacked and modified. So it is important to incorporate security technologies to protect the virtual properties in the MMORPGs.
- **Little and/or Weak Evidence:** This is caused by the above factors since both hard tracing and weak security protection result in systems in which there is not enough strong evidence for online gaming crime cases.

3.2 Security Requirements

According to our analysis of virtual property security above and our analysis of online gaming security issues [3, 4], there are several basic security requirements for virtual property protection, as follows.

- **Efficiency:** Log event tracing related to the virtual properties should be efficient. This requires efficient, yet accurate message logging.
- **Important Events Recording:** Important events related to the virtual property must be logged in a detailed and accurate fashion. For instance, agreements related to trading or transfer of property, wherein the trading agreement must contain detailed information, such as previous owner and current owner.
- **Signature:** Some important events related to virtual property may require non-repudiation security protection (e.g., a trading or releasing agreement).
- **Encryption:** Some log event records related to the virtual property may contain sensitive personal information (e.g., credit card account). Appropriately applied encryption technologies would provide protection for personal data, as well as resist malicious attempts to alter records in general.

4 Virtual Property Management Language

Virtual property management language is designed to offer efficient management for the virtual property through the recording of important events related to the virtual property with security mechanisms.

4.1 VPML Core Model

Our VPML core model consists of 7 entities: agreement, ownership, offer, event, ownership exchange, signature, and encryption. Figure 2 depicts the core model of VPML and the relationship among the entities.

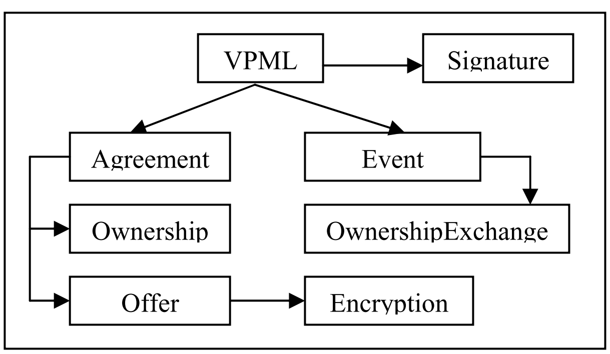


Fig. 2. VPML Core Model

In the VPML core model, the agreement entity expresses the current detailed trading information about the virtual property, for instance, the previous owner and the offer. In order to achieve higher security protection, the agreement message may require digital signatures from both the previous and current owners. The ownership entity contains the virtual property, previous owner, current owner, and ownership exchange method. The offer entity contains the price, trading time, and payment method related to the agreement. Some information like payment method in the offer entity may be sensitive for the customer and thus may require encryption protection. The event entity expresses the key historical events about the virtual property, including the ownership exchange events. In order to make VPML simple and more efficient, the information contained in the ownership exchange entity is the information about the previous agreement. In addition, this also strengthens system security. Detailed information about the core entities are described in the following sections.

4.2 VPML Agreement Model

VPML agreement model expresses the current agreement for the virtual property made by both previous and current owners. The agreement entity is an aggregation of two other entities as follows.

- Ownership – the information about the unique identity, previous owner and current owner of the virtual property, and ownership exchange method. The ownership exchange method could be trading, releasing, pick up, original, and others;
- Offer – the information about the trading related to the virtual property including the price, trading time, and payment method.

The agreement model provides evidence proving the current ownership of the virtual property. The signature entity in VPML, containing both the previous owner's signature and current owner's signature, would offer good security protection for the agreement.

4.3 VPML Offer Model

The VPML offer model expresses the latest trading information of the virtual property contained in the agreement. The offer entity is an aggregation of four other entities as follows.

- Price – the price information of the virtual property in this offer;
- Time – the time that the offer was processed and signed;
- PaymentMethod – the payment method information in the offer (e.g., credit card);
- Encryption – the information related to the encryption including encryption algorithm, key information and cipher data.

The offer entity provides the detailed trading information for an agreement. The use of encryption technology can give a better security protection for the personal data of the owners, especially payment information.

4.4 VPML Event Model

The VPML event model expresses important historical events regarding the virtual property. Currently, the event entity only contains the ownership exchange event. Figure 3 depicts the event model. The ownership exchange entity is an aggregation of three other entities as follows.

- Ownership – unique identity information about the previous owner and current owner of the virtual property in this ownership exchange event;
- Offer – detailed trading information about the virtual property in this ownership exchange event;
- Signature – information about the signature including signature algorithm, key information, and signature value.

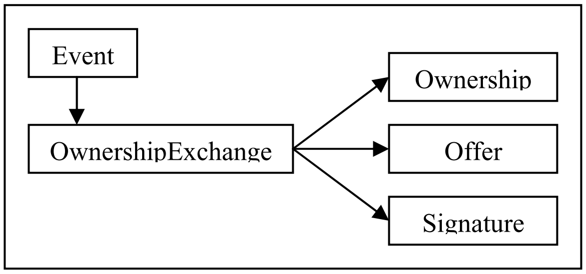


Fig. 3. VPML Event Model

The event model provides extra information for tracing the important events in the life of the virtual property. All information contained in the ownership exchange entity comes from the previous agreement, so the system only needs to put the previous agreement information into the event entity to create event records. This approach requires little extra in the way of computational resources.

4.5 VPML Encryption Model

The VPML core model contains signature and encryption components. We can use the same model as used in Open Digital Rights Language (ODRL) [10] for VPML encryption but we give a simple model for our approach, as follows. The encryption entity is an aggregation of three other entities.

- EncrytionMethod – encryption algorithm used in the encryption entity, e.g. 3DES;
- KeyInfo – key information or value used for the encryption, a session key value may require an encryption with public key system;
- CipherValue – encrypted data with the above encryption algorithm and key.

The encryption entity provides confidentiality protection for the sensitive personal data in VPML such as credit card account. In addition, the agreement signature is only on the cipher data so that the decryption is not required for verification. This makes VPML more efficient and secure since only the seller knows the payment info.

4.6 VPML Signature Model

The VPML signature model is an important model for the virtual property protection. As with VPML encryption, we can use the same model as used in Open Digital Rights Language (ODRL) [10] for VPML signature. We provide a simple model for it here. The signature entity is an aggregation of five other entities as follows.

- DigestMethod – hash algorithm used in the signature entity, e.g. SHA-1, SHA-256;
- DigestValue – hashing result with the hash algorithm;
- SignatureMethod – signature algorithm used in the signature entity, e.g. RSA;
- KeyInfo – public key certificate used for the signature;
- SignatureValue – signature result with the above signature algorithm and key.

The signature entity provides non-repudiation protection for the trading agreements in VPML so that any player can check if the owner is a real owner of the virtual property.

5 VPML XML Examples

In this section, we introduce two scenarios to explain how the XML syntax works for VPML.

5.1 Scenarios

Scenario 1: This scenario describes a general situation that a virtual property "sword of miracles" is distributed to the user named Steve by the system when Steve creates his account. The XML encoding of this scenario is described as follows (See Figure 4). In the coding, vid is the unique identity number of the virtual property, uid_p is the unique identity number of the previous owner of the virtual property, and uid_c is the unique identity number of the current owner.

```
<vpml>
  <agreement>
    <ownership>
      <vid>v8758947389/sword of miracles</vid>
      <uid_c>u8374847294/Steve</uid_c>
      <uid_p>u0000000000/System</uid_p>
      <exchangemethod>original</exchangemethod>
    </ownership>
    <offer>
      <price>0</price>
      <time>8:26/24/03/2003</time>
    </offer>
  </agreement>
  <signature>By System</signature>
</vpml>
```

Fig. 4. Original Agreement of "Sword of Miracles"

In this scenario, the payment method element in the offer entity and the current owner's signature for the agreement is not necessary since the offer price is zero and no reasons for the current owner to deny it. However, the system's signature is important for the current owner since the owner can claim the current ownership by the signature if something happens later, e.g. the virtual property is stolen. In addition, the signature technology is described in the next Section.

Scenario 2: In this scenario, Steve first released his virtual property to the system after playing one month for some reason (for instance, his container does not have enough room for this virtual property). Later, Peter finds the virtual property during his play and picks it up from the system. Peter later sells the virtual property to David for $800 after playing one year. Figure 5 depicts the XML encoding of this scenario. In the scenario, the current agreement is signed by both previous owner, Peter, and current owner, David, during trading. They can show the signature to a court if one of them denies the trading later. Furthermore, in order to make the system more efficient, we only store the previous owner's signature data in each owershipexchange entity of the event entity but this does not reduce the security of the applications since what we want is to trace the previous owner of each ownershipexchange from the event history and provide the evidence (signature) proving that the owner signed the agreement for the ownership exchange.

```
<vpml>
  <agreement>
    <ownership>
      <vid>v8758947389/sword of miracles</vid>
      <uid_c>u7389743894/David</uid_c>
      <uid_p>u3874238974/Peter</uid_p>
      <exchangemethod>trading</excahgnemethod>
    </ownership>
    <offer>
      <price>$800</price>
      <time>13:24/28/06/2004</time>
      <paymentmethod>Encrypted Information
      </paymentmethod>
    </offer>
  </agreement>
  <event>
    <ownershipexchange>
      <ownership>
        <uid_p>u0000000000/System</uid_p>
        <uid_c>u3874238974/Peter</uid_c>
        <excahngemethod>pick up</exchangemethod>
      </ownership>
      <offer>
        <price>0</price>
        <time>11:22/29/04/2003</time>
      </offer>
      <signature>By System</signature>
    </ownershipexchange>
    <ownershipexchange>
      <ownership>
        <uid_p>u8374847294/Steve</uid_p>
        <uid_c>u0000000000/System</uid_c>
        <excahngemethod>releasing</exchangemethod>
      </ownership>
      <offer>
        <price>0</price>
        <time>13:24/28/04/2003</time>
      </offer>
      <signature>By Steve</signature>
    </ownershipexchange>
    <ownershipexchange>
      <ownership>
        <uid_p>u0000000000/System</uid_p>
        <uid_c>u8374847294/Steve</uid_c>
        <excahngemethod>original</exchangemethod>
      </ownership>
      <offer>
        <price>0</price>
        <time>8:26/24/03/2003</time>
      </offer>
      <signature>By System</signature>
    </ownershipexchange>
  </event>
  <signature>By David</signature>
  <signature>By Peter</signature>
</vpml>
```

Fig. 5. History Records of "Sword of Miracles"

5.2 Security Mechanisms

The security technologies used in VPML include symmetrical key encryption technology and public key technology. We don't provide new algorithms for these security technologies but we describe how to embed existing security technologies into VPML to provide the protection for the virtual property and owners.

Symmetrical Key Encryption Mechanism: In VPML, some information like credit card account for payment is sensitive for the buyer. The buyer does not want other people, except the seller, to know it. VPML uses existing symmetrical key encryption technology to protect the buyer's personal data. The protection mechanism is described as follows. The buyer encrypts the sensitive information with a random one-time session key and makes an electronic envelope by encrypting the session key with the seller's public key. Thereafter, the buyer puts the encrypted information into the payment element and makes an offer to the seller. Figure 6 depicts an XML example for detailed encryption information for Scenario 2, above. Upon the reception, the seller decrypts the payment information and signs the agreement to the buyer. At the same time, the seller gets the payment from the buyer and delivers the virtual property to him.

Signature Mechanism: To protect the virtual properties in MMORPGs, VPML bundles the virtual properties with their agreements and event history using the signature mechanism. The signature verification will fail if other players want to change the ownership of the virtual properties illegally. Figure 7 depicts an XML VPML signature example provided by David in Scenario 2, above. The MMORPG system can audit the virtual properties under a player's account by verifying the VPML signature.

Key Management: The VPML security mechanisms include symmetric key technology and public key technology. We use different key management for them in order to make the system more efficient. For symmetric key technology, we use an electronic envelope for session key management. The detailed information is shown in the above symmetric key encryption mechanism. For the public key technology, we have many choices. One of them uses public key certificate such as X.509 [11]. Another uses the pseudonym technologies [12]. The public key management and pseudonym technology are beyond the scope of this paper.

```
<paymentmethod>Master Card
  <cardinfo>
    <encryption>
      <encryptionmethod> Algorithm= "http://www.w3.org
            /.../xmlenc#tripledes-cbc" </encryptionmethod>
      <keyinfo>
        <encryption>
          <encryptionmethod> Algorithm="http://www.w3.org
                        /.../xmlenc#rsa" </encryptionmethod>
          <keyinfo>
            <X509Data>
              <X509SKI> Seller's Public Key </X509SKI>
            </X509Data>
          </keyinfo>
          <ciphervalue>Encrypted Session Key </ciphervalue>
        </encryption>
      </keyinfo>
      <ciphervalue>EncryptedCreditCardInfo</ciphervalue>
    </encryption>
  </cardinfo>
</paymentmethod>
```

Fig. 6. XML VPML Encryption Example

```
<signature>
  <digestmethod>
    Algorithm="http://www.w3.org/.../smldsig#sha1"
  </digestmethod>
  <digestvalue> Hashed Value </digestvalue>
  <signaturemethod>
    Algorithm="http://www.w3.org /.../xmldsig#rsa-sha1"
  </signaturemethod>
  <keyinfo>
    <X509Data>
      <X509SKI> David's Certificate Info</X509SKI>
    </X509Data>
  </keyinfo>
  <signaturevalue> Signature Value </ciphervalue>
</signature>
```

Fig. 7. XML VPML Signature Example

6 Conclusions

Online gaming crime is becoming a serious issue for our society. In some countries, many criminal cases are related to virtual property according to the latest research and statistics [3, 4]. In order to provide secure, efficient protection and management for virtual properties in the online gaming systems, we propose a virtual property management language. VPML can satisfy some specific requirements of online gaming systems such as efficient tracing and flexible security protection. Some related security problems are not discussed in this paper such as PKI and its development. In this paper we have provided an overview of the VPML model with a description of the different entities and examples of records and agreements for two different scenarios using this model and XML.

For practice, the VPML file can be stored on the distributed client side and implemented in a separate program which has an open interface connect with the gaming software. However, the gaming center can store a copy of the VPML data in its database. This needs a further research.

References

[1] Global online games, Datamonitor Corp. November, 2002.

[2] Annual Criminal Statistics Report, National Police Administration, Taiwan, 2003.

[3] Y. Chen, P. Chen, R. Song, and L. Korba. Online Gaming Crime and Security Issue – Cases and Countermeasures from Taiwan. In Proceeding of the second annual conference on Privacy, Security, and Trust (PST2004), Fredericton, NB, Canada, October 13-15, 2004. NRC 47401.

[4] Y. Chen, P. Chen, R. Song, G. Yee, and L. Korba. Taxonomy of Online Gaming Crime and Security. In Proceedings of the 16th information resources management association international conference (IRMA2005), San Diego, California, USA, May 15-18, 2005.

[5] Celeste Biever, "Sales in Virtual Goods top $100 Million", NewScientist.com news. Available at "http://www.newscientist.com/news/news.jsp?id=ns99996601". Accessed in October 4, 2004.

[6] A virtual dragon-sky knife is sold on $4800 U.S. dollars in charity bazaar, United Daily, Taiwan, January 29, 2003.

[7] Everquest Online, SONY Corp. Available at "http://www.sony.com".

[8] Lineage II 2 US kain server custom 50 level char. Available at "http://cgi.ebay.com/ws/eBayISPI.dll?ViewItem&category=4596&item=8112795841&rd=1". Accessed in June 10, 2004.

[9] Itembay Taiwan Corp. Available at "http://www.itembay.com.tw". Accessed in June 20, 2004.

[10] Open Digital Rights Language (ODRL). Available at "http://www.odrl.net/".

[11] CCITT Rec. X.509 (03/00) | ISO/IEC 9594-8: 2001, Information Technology – Open Systems Interconnecion – The Directory: Public-key and Attribute Certificate Frameworks.

[12] Lysyanskaya, A., Rivest, R. L., Sahai, A., & Wolf, S. Pseudonym Systems. In Howard Heys and Carlisle Adams (Ads.), SAC'99, LNCS 1758, pp. 184-199, 2000.

The Research and Implementation of Semantic Based RDF Tagging and Webpage Searching Web Service

*Jason C. Hung, **Schummi Yang, **Mao-Shuen Chiu and **Timothy K. Shih
*Department of Information Management
Northern Taiwan Institute of Science and Technology
No. 2, Xueyuan Rd., Peitou, 112 Taipei, Taiwan, R.O.C
**Department of Computer Science and Information Engineering
Tamkang University
Tamsui, Taipei Hsien, Taiwan, R.O.C.
E-Mail:692192148@s92.tku.edu.tw

Abstract

In recent years, the network and information technology is more and more ripe we always search for data or files through the Internet. The resources on the World Wide Web are increased day by day. The amount of information exchange and speed of update are also grown. Many methods depend on the match of vocabularies between information request and searched objects. For example, we usually adopt some keywords and Boolean operators to form the query to search the Internet for interested information. Unfortunately, users do not always use proper words and operators to form the query for the search. The result is related or unrelated information is both retrieved .The retrieval method becomes critical for us to get more accurate results. In order to improve the searching process, a RDF-based mechanism integrated with word sense disambiguation technique is proposed to semantically index and retrieve web pages on the World Wide Web. The approach is to describe the resources by RDF(S) metadata and store them. The proposed method also has additional advantage of the ability of further integrated into the Semantic Web Service.

Keywords: Semantic Web, WordNet, RDF, Jena, Web Service.

1. Introduction

A large amount of information is retrieved from the Internet by using search engines. A search engine requires one or more user input keywords to carry out a search, but sometimes the search results do not match expectations from users due to the huge amount of information can be accessed through Internet. Network information technology is unceasingly developed and mature, when we want to seek for some information, we often search the WWW. The resources on the network are increasing day by day, and the information exchange and absorption rate are also multiplying. The search engine, like Google, has three hundred million inquiries and 40 hundred million indexed homepages every day in the World Wide Web. The automated searching usually uses the web crawler, spider, robot (bot) or the agent technology, follows the HTTP hyperlink between websites to search and collect web pages on the Internet.

The W3C **[4]** research team proposed the RDF (Resource Description Framework) standard **[8]**. RDF is based on XML **[9]** and in XML grammar foundation. RDF stipulates the metadata storage structure and related technical standard. Using the RDF language, we can characterize the information in a uniform, exchangeable style. Furthermore, this makes it possible for machines to "understand" the data.

If the search engine searches the Internet depending on the conceptual matches, not just relying on the similar word usage, it would have better search ability to respond the detail inquiry request. Using RDF tagging may provide opportunity for new search methods. But, the majority of semantic search engine will meet the potency problem when searching for information within the massive semantic network. In order to obtain the effective searching results, the network must contain massive related information. But meanwhile, the large-scale network would cause to discover a best way in many processing solutions question way to appear is not easily.

However, the Internet search has some problems. For example, when using nowadays search system, we always utilize keywords to query. But to make the inquiry in the material huge information sea, we often fall into another information sea, even more information sea. Thus we may find more unrelated data than truly needed data, the searching efficiency will thus reduce very much. The cause of this condition is because the words meaning are confused or the improper resources description or tag. We knew that it is an issue. In general, much of the data in World Wide Web has less correlation defined between each other, and this makes it difficult to search related material from one to another.

We proposed a mechanism to achieve the semantic web service system. It includes the semantic searching method, combines WordNet **[2]** and RDF to search the different characteristic resources from the Internet. The goal is to supply interaction and commutation between heterogeneity resources, to annotate them by RDF and WordNet metadata, and to supply other agent process to inquiry these RDF tagged documents. It achieved semantic web page resources sharing progress for human natural operation and convenience.

Therefore, according to the concept-based WSD system model proposed by Che-Yu Yang (AINA'2005) **[1],** study of this paper will focus on using RDF description along with WordNet **[3]** Synset (synonymy word collection) to describe/tag semantic meaning of web-pages content, as well as using the characteristic – Notation Triples (N3) – to give signs for marking these various resources with other resources. The query language for RDF documents is Jena **[6]** which provides N-Triple to search the words relative. Finally we established the Semantic Web Service, achieved the goal to supply the semantic query and sharing resources.

This paper is organized as follows. In section 1 we give a brief introduction about the research background and our system. We talk about the searching advantage of our idea. Next, in section 2 is related work about the RDF and its query language. In section 3 we discuss about the relationship between RDF and the WordNet. We established a method to add the semantic concept into web pages that can help the process of search and query. In section 4 we integrate the whole system and package it as Web Service. Finally is the conclusions and future work in section 5.

2. Related Work

The Resource Description Framework (RDF) is a W3C Recommendation for the formulation of metadata-description on the World Wide Web. The RDF is a simple model and considered to be the most relevant standard for data representation and exchange on the Semantic Web **[10]**. The RDF Schema (RDFS) extends this standard with the means to specify domain vocabulary and object structures in order to describe and define grammar as well as the announcement the RDF. RDFS looks like a dictionary, it describes each property significance, the characteristic, and the constraint of property value. RDFS may let the person to read and understand each data attribute significance. RDFS defines the class and property to describe the resources content.

Also RDF is for knowledge and metadata representation. And it is as well fitting for representing any data or metadata. RDF may be regarded as one kind of Web knowledge to express the language, or said it is a logical language. RDF has the formalized grammar, the semantic model, the ability to prove inference as well as and theorem of the reliability. The architecture of RDF is based on the Extensible Markup Language (XML). Therefore we may use RDF data model and use the directional characteristic to construct the relation. So we will use RDF to describe metadata.

We integrated the RDF Query Language (RDQL). RDQL is used to query RDF documents language in the tradition of database. RDQL is a typed language for generalized path expressions featuring variables. In order to query the RDF documents, we use the Jena's API architecture to focus on the RDF data model. Jena is Java toolkit for developing semantic web applications based on W3C recommendations for RDF and OWL. It provides an RDF API; ARP, an RDF parser, RDQL, an RDF query language; an OWL API; and rule-based inference for RDFS and OWL. A basic RDF/XML document is created by instantiating one of the model classes and adding at least one statement (N3) to relate them for accessing the metadata or elements attribute.

We also utilized the Sesame **[7]**. Sesame is an open source RDF database with support for RDF Schema inferencing and querying. Sesame is a Java framework for storing, querying and inferencing for RDF and RDFS **[12]**. It can be deployed as a web server or used as a Java library. Features includes that allow persistent storage of RDF data and schema information and subsequent querying of that information.

3. Design and Functional Requirements for the RDF Storage and Query

As the extension of the concept-based word sense disambiguation (WSD) model which was proposed by Che-Yu Yang **[1]**, this paper will focus on using RDF format, combined with WordNet ontology, to indexes/tags web pages with concepts (actually the synonymy sets, usually called "synsets"). We also use the characteristic – Notation Triples (N3) to mark these various resources with other resources constructions.

When users want to query something by the keywords, through the WSD module **[1],[10]** and Google API **[5],** we can filter out semantically unrelated web pages (miss-retrieved) and leave only conceptually matched web pages. Next, we use the synonymy sets (synsets) in WordNet with the N3 notation in RDF to annotate the keywords on the web pages with their own concepts/senses in the context. When users use keywords to search the Internet, the keywords will be disambiguated by the WSD module and assigned/tagged with each WordNet's synset-id to each keyword according to their meanings/senses in the context. Not only the keywords in the user query, but also the keywords in the conceptual matched web pages after disambiguation are tagged/annotated with synset-ids. That's actually the semantic mapping between keywords in web pages and senses/concepts in WordNet, as shown in the figure 1.

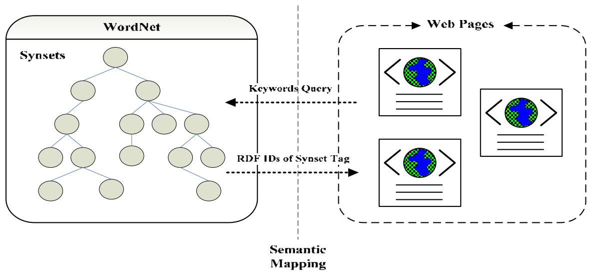


Figure 1. Semantic mapping between WordNet and web pages

We can use N3 notation (subject, predicate and object) of RDF to achieve the above idea. Using N3 notation we can establish the relation between the concepts/senses in Wordnet and keywords in web pages to archive the RDF/RDF(S) characteristics. And we construct the N3 notation by the synset-id and URI (rdf:alt), as shown in figure 2. It represents the properties and attributes of the resources type. The word (book) is oriented towards the relative web pages (URIs) and the arrow point means to the property each other(wn:). The "wn:" represents the Wordnet RDF schema.

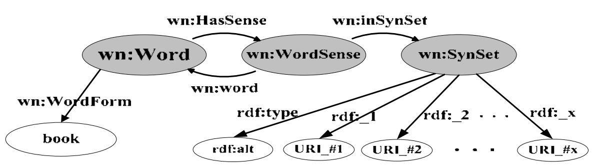


Figure 2. The semantic relations of the Wordnet in RDF

4. System Architecture and Implementation

We employ the glossaries defined in WordNet and in WordNet Schema, and use the synset-id of glossary itself to establish the connective model, and we also take shape representation of RDF Model. Because the webpage may consist of many different glossaries, we can combine these different glossaries with different URI of WordNet synsets. Finally they will form a hierarchical heterology structure RDF Model. We treat RDF as the reference structure of synset-ids. According to the different webpage resources characteristic we construct RDF Model Sets. The RDQL provides the ability to query the RDF annotated documents.

Before tagging documents with the RDF notation, we must make the index of them first. After parsing the RDF documents (the N-Triples part) and collecting correlative data to store in our database, we establish the entity-relation table to store the transformation results. The method is to treat each N3 (Subject, Predicate and Object) as the same identical unit when storing them. (This method offers more efficiency when we want to do concept searching and index catalog of contents service). This is due to the need of reference and index of the resources service in the future. The architecture diagram is as figure 3.

So, when users query with the keyword(s), the WSD algorithm will semantically filter out the unrelated web pages and leaves only semantically matched ones that are returned from the Internet search request from Google API. And we tag the matched keywords in the web pages with synset-ids when carry out semantic mapping between web pages and Wordnet. Then the system will establish the related property sets which includes the attributes of the subject, predicate and object. They are the classes of words named by URIs. To reference the RDF/RDFS format storage proposed in our system. Finally we package the whole procedure into a Semantic Web Service.

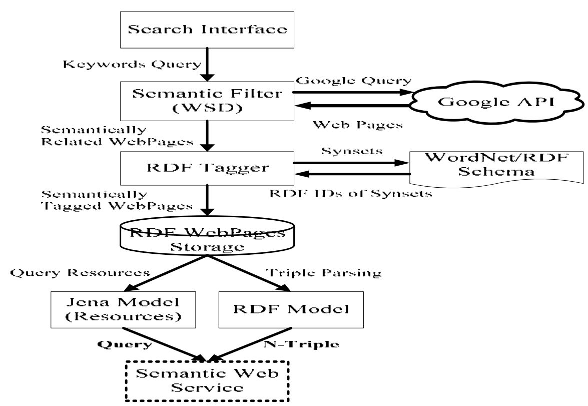


Figure 3. The system architecture

Jena is the standard inquiry language of RDF and RDFS. It provides the diverse searching measures, including files or the website content. Also it can supply to limit the scope of the searching (use "WHERE" Clause), logical determining and filtering ("AND" Clause, "USING" Clause).The Jena development platform is based on Java Framework. It's the application program interface (API) which can construct the query layer of Semantic Web architecture.

We utilized the "Sesame", which supports Jena query language, to integrate the RDQL into our system. And we use "mysql" as the storage database, which includes the XML and RDF documents.

When user puts a query, the system will access the repository (Sesame) to search the RDF documents by Jena API in our database (mysql). According to the properties of N3 we know that the characteristics of the subject, predicate and object. Figure 4 is our system flow chart.

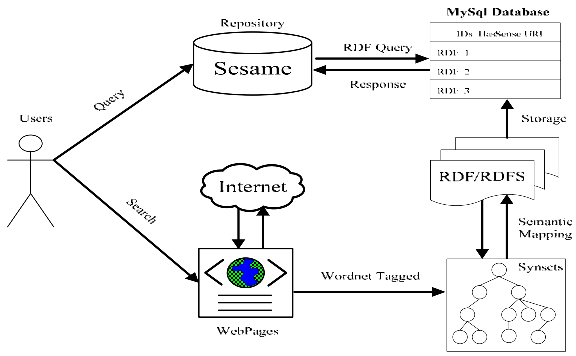


Figure 4. The system flow chart

We construct the Web Service system by Java Web Services Developer Pack (Java WSDP) **[12]**. The environment of our implementation is the Apache Jakarta Tomcat (Apache SOAP 2.0, Java servlets 2.2 API standard), and we installed the Apache Xerces XML Parser 1.2.3. It supplies the XML grammar parsing and supports to develop the most XML standard. Figure 5 is our system interface, as follows.

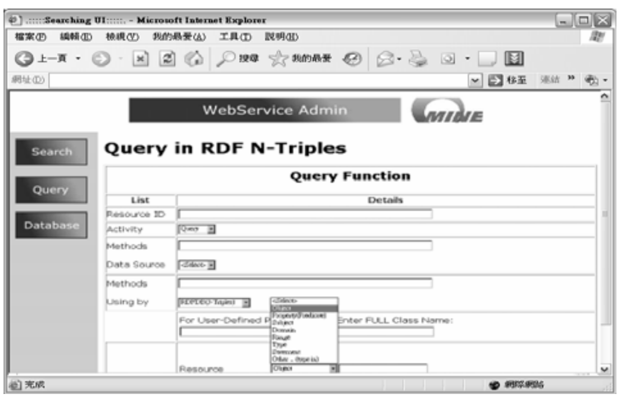


Figure 5. Web Service system

We proposed the semantic Web Service system, including the search and query method to the different resources by the attribute characteristic in WordNet and RDF. Web Service can communicate with different users. Our system achieved the resource sharing, exchanging and communication as well. The goal is the resource sharing to achieve both the human and machine operation convenience.

5. Conclusions and Future Work

The system based on semantic will mark up from the different resources. It offers the ability to search and query the resources on the World Wide Web, finally packs into Web Service system to improve systematic practicability further. We propose the system to improve on searching more semantic and sharing resources.

We will discuss in the study on Ontology in the future. The metadata is packaged up Ontology to integrate relative object. In addition, the OWL (Web Ontology Language) is a component of the Semantic Web framework. OWL is built upon RDF and RDF Schema. OWL also adds more vocabulary for describing properties and classes. It's intended to provide a language that can be used to describe the classes and relations between them that are inherent in Web documents and application.

References

[1]. Jason C. Hung, Ching-Sheng Wang, Che-Yu Yang, Mao-Shuen Chiu, George Yee, "Applying Word Sense Disambiguation to Question Answering System for E-Learning", International Conference on Advanced Information Networking and Applications (AINA 2005).

[2]. George A. Miller, "WordNet: A Lexical Database," Comm. ACM, Vol. 38, No. 11, 1993, pp. 39-41.

[3]. Leacock and Chodorow. 1998. Combining local context and WordNet similarity for wordsense identification. In Fellbaum 1998, 265-283.

[4]. W3C, World Wide Web Consortium, http://www.w3c.org/

[5]. Google API, http://www.google.com.tw/apis/

[6]. Jena, http://jena.sourceforge.net/

[7]. Sesame, http://openrdf.ord/

[8]. Tim Berners-Lee, 2000. "Primer: Getting into RDF and Semantic Web using N3".

[9]. M.Klein, "XML, RDF, and Relatives," IEEE Intelligent Systems, vol. 16, no.2, Mar/Apr 2001, pp. 26-28.

[10]. Jiangsheng Yu,"WSD and Closed Semantic Constraint" supported by National Foundation of Natural Science (Research on Chinese Information Extraction) No. 69483003 and Project 985 in Peking University.

[11]. Boris Katz Jimmy Lin, "Annotating the Semantic Web Using Natural Language, " In Proceedings of the 2nd Workshop on NLP and XML (NLPXML 2002) at COLING 2002, September 2002, Taipei, Taiwan.

[12]. Java Web Service Developer Pack (Java WSDP) http://java.sun.com

A Planning Approach to Media Adaptation within the Semantic Web

Matthias De Geyter and Peter Soetens
VRT
Brussels, Belgium
{matthias.degeyter, peter.soetens}@vrt.be

Abstract

The continuous increase of multimedia content on the Web and different device types that have access to this content, demands for an extensible approach to media adaptation. To deliver highly adapted content to a specific device, a flexible engine will need to understand the device's constraints and adapt accordingly. In this paper, such an adaptation framework relying on Semantic Web and AI planning technologies is proposed. Semantic Web mechanisms like OWL and OWL-S enable us to describe the parties involved in the adaptation process: a multimedia document, adaptation components and a target device. Based on the available descriptions, a planning algorithm will create a chain of adaptation components, each modifying the original document until the result is suitable for publication on the target platform.

1. Introduction

The booming amount of multimedia files and formats offered to the consumer on the Web, next to the continuous emergence of new devices that have access to this content, calls for a new approach towards media adaptation. The creation of publications tailored to the specific needs of different client platforms such as mobile phones and personal digital assistents, personal computers, televisions with settop boxes –each with their own requirements and preferences regarding e.g. bandwidth and resolution– becomes a very tedious and increasingly difficult process. Traditional multimedia distribution mechanisms are mostly either completely unaware of these different specifications, or allow only for a very limited set of adaptations, such as low-quality and high-quality versions depending on the client's bandwidth. Nevertheless, the tools to modify media documents are available, providing several operations such as the transmoding and transcoding of presentations. The challenge lies in an efficient and flexible architecture for such adaptations; if different combinations of source documents and clients required the execution of completely distinct programs, this could result in a vast amount of media processing applications needed by a provider wanting to address the broad range of platforms nowadays available.

The architecture we propose makes use of adaptation components that perform basic operations, and as such can be regarded as the building blocks for processing media. By chaining these software components, complex adaptation scenarios become possible. Web Services are a logical choice for these basic elements, because of their flexible and distributed nature. In this way, it is quite easy to enlarge the set of possible operations when new media formats or platforms arise. The main problem still to be tackled, is the selection and correct ordering of the right Web Services. Some illustrative basic components in our scenario are video encoding, decoding and scaling operations. As an example, we can imagine the case where the client platform requests a scaled version of a certain video file, but using a different codec. The source video file is e.g. MPEG-2, while the target platform requests a spatially scaled MPEG-4 file. If the scaling service can't take MPEG-2 as an input, but demands for raw YUV, then our system has to automatically apply decoding before scaling, finally followed by reencoding.

Therefore, descriptions of the source document, the client platform as well as the available adaptation components are necessary. In [11] we proposed a single domain ontology in the Web Ontology Language (OWL) to annotate the parties involved. Furthermore, an intelligent adaptation strategy that selects the appropriate services on the right time, is needed. A first implementation of such a strategy was presented in [10]. In this paper, we discuss a refined approach utilizing a formal planning strategy to increase the genericity of our framework.

The outline of this paper is as follows: first, in sect. 2 we describe some existing approaches to the multimedia adaptation problem. Next, in section 3 the Semantic Web technologies involved in our approach are introduced, while section 4 covers possible adaptation strategies and digs into our own implementation.

2. Approaches to Multimedia Adaptation

Many media adaptation techniques propose new document models or combine existing ones to support interactivity and media items such as text, images, audio and video. The Cuypers engine [12] has a somewhat different approach: rather than adapting existing presentations, the publications created by the Cuypers engine are based on an abstract semantical model of the presentation which is more detailed throughout the adaptation process, in a hierarchical way. According to Boll [1], this system would demand additional programming when complex presentation generation tasks are needed. She proposes a programming framework with interfaces to develop personalized applications. In this way, one can integrate existing mechanisms such as MPEG-21 or CC/PP to incorporate platform capabilities and user preferences. A programming framework however, will not provide the flexibility needed in the rapidly evolving world of multimedia on the Internet. In this context, Web Services offer greater independence from programming language or network topology and thanks to the introduction of an appropriate description language, their dynamic composition based on domain knowledge becomes possible. In most of the relevant research on this topic [5] [2] [8], a technique is used which couples the description of a service with a description of a planning or task methodology.

Applying this approach to the problem of multimedia adaptation demands for descriptions of the three parties involved, namely the source document, the adaptation services and the target device. Within MPEG-21, DIA (Digital Item Adaptation) aims to provide a standard to describe the metadata needed when performing adaptation operations, such as device capabilities. However, this vocabulary is difficult to extend and no formal definition of the semantics of MPEG-21 terms is given. Problems arise when one wants to build an MPEG-21-based adaptation engine [4] with document descriptions based on MPEG-7, since such an approach requires mapping schemes between the MPEG-21 and MPEG-7 terms. The introduction of new media formats and devices necessitates the modification of these mappings and the adaptation component descriptions, thus limiting the flexibility of this approach. Furthermore, a richer semantical model would extend reasoning possibilities, preventing the problems with planning heuristics described in [4]. For this reason, we introduced several Semantic Web technologies in our adaptation framework, as explained in the next section.

3. The Semantic Web and Media Adaptation

3.1. The Semantic Web

The Semantic Web was thought up by Tim Berners-Lee, inventor of the World Wide Web. In essence, the Web is nothing but an information space. However, it should evolve into something not only useful for human-human communication, but also for machines, enabling them to participate and help. One of the major obstacles to this is the fact that most information on the Web is designed for human consumption, and even if it is derived from a database with well defined meanings, the structure of the data is not evident to a robot browsing the web. Instead of training machines to behave like people, the Semantic Web approach develops languages for expressing information in a machine processable form, which is far more feasible than the former. The Semantic Web is generally built on syntaxes which use URIs to represent data, usually in triple based structures using a set of particular syntaxes developed especially for the task. These syntaxes are called Resource Description Framework (RDF) syntaxes. In this manner, one can write down statements about data or other statements. RDF Schema is used to define properties and classes, along with constraints on how to use them. Such a schema layer makes it possible to check the semantic correctness of documents.

OWL [13], the Web Ontology Language, is an additional layer which builds on top of RDF Schema. OWL facilitates greater machine interpretability of Web content by providing additional vocabulary along with a formal semantics. It allows for richer expressions (relations between classes such as disjointness, and cardinality restrictions among other things) and can be used to represent ontologies, which are generally defined as a "representation of a shared conceptualisation of a particular domain" [3]; a description of the concepts and relationships between those concepts in a certain domain, enabling knowledge sharing and reuse.

In parallel to this evolution towards the Semantic Web, the Web is also evolving towards being a provider of services that can communicate with each other: web sites do not merely provide static information but allow one to effect some action or change in the world (e.g. flight booking services). In order to employ their full potential, appropriate descriptions for Web Services need to be developed. Current technologies such as UDDI, WSDL and SOAP provide limited support for important tasks like service discovery, composition and monitoring [6]. To enable a reliable, large-scale interoperation of Web Services, machines should be able to help in the selection process of the appropriate services. This is were the Semantic Web and Web Services come together: The Semantic Web promises services whose characteristics are encoded in an unambiguous, machine understandable form.

OWL-S is an ontology building on top of OWL, to describe Web Services using a service profile and process model. The transformation produced by the service is specified in terms of Inputs, Outputs, Preconditions and Results (IOPRs). Preconditions and Results denote external conditions required by the service and different effects –together with associated outputs– resulting from different conditions during the execution, respectively. Composition of several services to achieve one final goal can be performed by matching their respective IOPRs.

3.2. Apply the Semantic Web to Media Adaptation

For the work on a news distribution system for Belgian broadcaster VRT, we developed a single media domain ontology in OWL, which is used to describe the source documents, the target platforms and the adaptation services. The ontology is modelled after elements from both MPEG-21 and UAProf, and allows us to express parameters such as the media type (e.g. still image or video) and its underlying characteristics like resolution and bit rate. We extended the RDF-based CC/PP framework to annotate the client's capabilities using our OWL ontology. Inputs and outputs of the available services, which mainly consist of video processing services such as scaling, decoding and encoding, are also defined in terms of these concepts. However, the expression of conditions (needed to write down Preconditions and Results) demanded for the introduction of an appropriate formalism. For this, we were inspired by the DRS (Declarative RDF System) and SWRL (Semantic Web Rule Language) vocabularies, which originated because RDF and notations based on it such as OWL, are not very suited to express arbitrary propositions –like Prolog-style Horn clauses– needed to represent processes and plans. We developed a mechanism that allows us to make simple assertions about the values of OWL Properties, also permitting mathematical expressions (see fig. 1 for an example).

```
<xmt:XMTExpression rdf:ID="CorrectInputFormat">
 <xmt:hasExpressionFormula>
  <xmt:Formula rdf:ID="InputFormatFormula">
   <xmt:hasSubject rdf:resource="#InputVideo"/>
   <xmt:hasProperty rdf:resource="
    http://lt.xmt.be/domainontology.owl#hasVideoFormat"/>
   <xmt:hasDescription rdf:datatype="
    http://www.w3.org/2001/XMLSchema#string">
    http://lt.xmt.be/Scale.owl#SupportedCodecs
   </xmt:hasDescription>
  </xmt:Formula>
 </xmt:hasExpressionFormula>
</xmt:XMTExpression>
```

Figure 1. Example of the formalism used to express conditions

4. Multimedia Service Composition

4.1. Basic Concepts of the Composition

Next to appropriate descriptions of source documents, available adaptation components and client platforms, based on technologies discussed in the previous section, a system to pick the right services in the correct order is needed. Therefore, the matching of media document and client properties is required, revealing which characteristics of the document have to be adapted to comply with the client's demands. Such a property matcher makes use of a reasoner that detects equivalence and subsumption relations between concepts in our OWL ontology.

In our first implementation [10] an algorithm that selects services based on their IOPR-descriptions was implemented. For example, if the media description holds a property resolution, whose value needs to be changed according to the client specifications, then services affecting this feature in their result statements become possible for execution. Services are organized in a service registry, where they are categorized conform to the IOPRs, in order to confine the search space and speed up the selection process. If a chosen service can't be executed because of its preconditions, the objectives are temporarily altered to solve this problem first. Otherwise, the OWL-S effects of this service are calculated to update the media characteristics. The algorithm iterates until the media characteristics fit the client's demands, or no solution is found after a fixed maximum number of iterations. This implementation has a few shortcomings, though. First of all, there are no backtracking possibilities, which means that the initial property picked as a basis for service selection is quite important, especially in more complex scenarios. Secondly, certain boundaries taken into account when comparing values and evaluating expressions, were coded into the algorithm. In order to achieve a more generic adaptation engine, these should shift towards the domain ontology. This could be accomplished by means of the same formalism we used to express conditions in section 3.

4.2. Planning for Web Service Composition

For the purpose of increasing genericity, we studied some existing AI planning methodologies to perform the composition. One of the most popular techniques applied in automatic generation of Web Service composition plans [9, 7] is Hierarchical Task Network (HTN) planning, since its concept of task decomposition is very similar to the concept of composite process decomposition in the OWL-S process ontology. Typically, an HTN planner starts with an

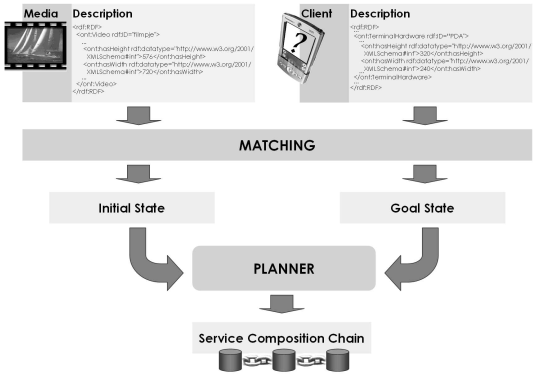


Figure 2. Schematic representation of multimedia service composition through planning

```
FINDPLAN(currentplan π,goals g,services A, states S)
    if goals g=∅ then return plan
    repeat
        A ← {a | a ∈ A and a is a ground instance of an operator in O
        and a is relevant for g and a ∉ S and
        a is not tried before at this stage of the plan }
        if A=∅ then return failure
        nondeterministically choose any action a in A
        if a has preconditions :
            Π'=FINDPLAN(π,preconditions(a),recompute(A), S)
            if Π'≠ ∅ for every plan π'∈ Π'
            s=last state in S(=set of states of plan π')
            s' ← γ(s,π')
            s' ← γ(s',a)
            S=S∪{s'}
            π ← π.π'.a
            Π=Π∪FINDPLAN(π,recompute(g),recompute(A), S)
        if no preconditions :
            s=last state in plan π
            s' ← γ(s,π')
            s' ← γ(s',a)
            S=S∪{s'}
            π ← π.π'.a
            Π=Π∪FINDPLAN(π,recompute(g),recompute(A), S)
    Until A=∅ then return Π
```

Figure 3. Planning algorithm to determine the service chain

abstract network representing the task to be solved, and proceeds by expanding this abstract network into more detailed subtasks until the networks contain executable actions. The description of a planning domain includes a set of basic operators –corresponding to the adaptation services in our particular case– and a set of methods, each of which is a prescription for how to decompose a task into smaller tasks. By encoding this knowledge on how to go about looking for a plan in the domain, the search space is greatly reduced. On the other hand, if we want the service composition to be fully automated, with minimum human intervention and as generic as possible, the methods should be generated in an automatic manner. Thus, a mechanism to transfer the domain knowledge captured in our OWL ontology to the appropriate HTN methods is necessary. Our experiments with the HTN planner JSHOP2 showed that this is a complex problem, since a lot of rules are required to achieve an efficient composition chain. One possible improvement would be to define composite OWL-S processes; in our current implementation every adaptation service consists of only one atomic process (e.g. decode, scale ...). Composite processes corresponding to likely adaptation combinations (e.g. a service that decodes a video and reencodes it in another format) could be mapped on the HTN methods. However, in a generic framework, it should be able to constantly add and remove basic services without relying on such predefined composite process descriptions. It is also mentioned in [9] that composition of services by HTN planners is better suited for the specialization of prewritten composite processes than for the creation of complex, novel programs (such as a media adaptation chain).

4.3. Implementation of a Planning Approach

Our first adaptation strategy was quite resemblant to a basic STRIPS algorithm, which also uses operators' preconditions as subgoals to reduce the size of the search space. Because of this hand-on experience, we implemented an extended version of such an algorithm which includes backtracking, after adapting our engine to support a smooth integration of various planning algorithms. The matching process now results in an initial state and a goal state, wherein all properties are taken into account (see fig. 2). Characteristics of the media document that need to change, constitute the planner's goals, while services that are relevant for one or more of these goals –or subgoals in the case of unfulfilled preconditions– are now selected in a nondeterministic way (fig. 3).

Based on the evaluation of the latest chosen service's preconditions and effects, new goals and relevant services are computed at each step to restrict the search space. This is repeated until all appropriate services at every step are tried. Furthermore, the limitation that every service can occur only once in a certain plan is imposed. Thus, only valuable plans are generated. Just like the algorithm explained in [10], this new implementation allows us to successfully adapt video to different platforms, but now multiple solutions are offered. The extension to more complex adaptation scenarios is relatively easy now, with the adjustment of

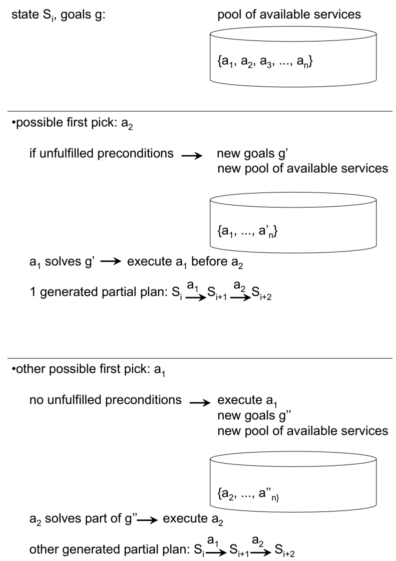


Figure 4. Under some circumstances, subplans may be generated more than once

the domain ontology as the main remaining task.

Optimization of the planning algorithm is still possible since some plans can be generated twice (see fig. 4 for an example): a service a_1 needed to fulfill the preconditions of the service a_2 picked at step i, could itself be among the available services at step i, and will therefore be chosen a next iteration. As a consequence, a certain subproblem would be solved twice, although prior conditions (relevant services still available) may differ. Nevertheless, caching of subplans could be useful. The introduction of an evaluation function to select best partial plans may also improve efficiency.

5. Conclusion and Future Work

Semantic Web technologies can help to create an efficient and flexible multimedia adaptation architecture. A domain ontology enables us to create semantically rich descriptions of source documents, client platforms and adaptation components. By matching the descriptions of a media document and a target platform one can derive an initial state and a goal state for a planning algorithm.

Since most planners use a notation different from our OWL knowledge representation, it was not possible to simply plug in existing planner implementations. Furthermore, unlike most typical planning problems, many of our services have several input parameters that influence the effect of the action. The problem is that in some cases this makes the result set, containing all possible outcomes of the action, no longer being discrete. Until now, we compute a service's input values solely based on the client specifications. Future work may involve computing a set of optimal input values for each service based on the description of its effects or on previous executions of the service.

Our experiments have shown that we can generate a correct service chain for many different media and platform descriptions. However, so far our ontology is restricted to the domain of video. Ongoing work involves extending it to other media types such as text and audio. This will enable us to perform more complex adaptations such as the adjustment of a document containing text and video to guarantee an optimal viewing experience. This could imply the scaling of the text and the video, where these two can have an impact on each other. By increasing the complexity of the adaptation scenario, we hope to offer further proof of the strength of our approach.

6. Acknowledgements

This work is the result of a joint collaboration between VRT, IMEC, Vrije Universiteit Brussel and Universiteit Gent, funded by the Flemish government.

References

[1] S. Boll. Mm4u - a framework for creating personalized multimedia content. In *Proceedings of Distributed Multimedia Systems.*, 2003.

[2] A. Gómez-Pérez and R. González-Cabero. A framework for design and composition of semantic web services. In *Semantic Web Services - 2004 AAAI Spring Symposium*, 2004.

[3] T. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[4] D. Jannach, K. Leopold, H. Hellwagner, and C. Timmerer. A knowledge based approach for multi-step media adaptation. In *Proceedings of 5th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, 2004.

[5] J. Kim and Y. Gil. Towards interactive composition of semantic web services. In *Semantic Web Services - 2004 AAAI Spring Symposium*, 2004.

[6] S. McIlraith, T. C. Son, and H. Zeng. Semantic web services. *IEEE Intelligent Systems*, 16(2):46–53, March/April 2001.

[7] M. Paolucci, K. Sycara, and T. Kawamura. Delivering semantic web services. In *WWW2003*.

[8] E. Sirin, B. Parsia, and J. Hendler. Composition-driven filtering and selection of semantic web services. In *AAAI Spring Symposium on Semantic Web Services*, 2004.

[9] E. Sirin, B. Parsia, D. Wu, J. Hendler, and D. Nau. HTN planning for web service composition using SHOP2. *Journal of Web Semantics*, 1(4):377–396, 2004.

[10] P. Soetens and M. De Geyter. Multi-step media adaptation: Implementation of a knowledge-based engine. In *WWW2005*, 2005.

[11] P. Soetens, M. De Geyter, and S. Decneut. Multi-step media adaptation with semantic web services. In *International Semantic Web Conference (ISWC 2004)*, 2004.

[12] J. van Ossenbruggen, F. Cornelissen, J. Geurts, and L. Rutledge. Cuypers: a semi-automatic hypermedia presentation system. In *Techn. Rep. INS-R0025*, 2002.

[13] W3C. *OWL Web Ontology Language Overview – W3C Recommendation*, Feb. 2004. http://www.w3.org/TR/owl-features/.

A Component based Multimedia Middleware for Content Production Factory

T. Martini, P. Nesi, D. Rogai, A. Vallotti

DSI-DISIT, Department of Systems and Informatics, University of Florence
Via S. Marta, 3 - 50139 Florence, Italy, nesi@dsi.unifi.it, http://www.disit.dsi.unifi.it

Abstract

Recently specific middlewares have been proposed while those with general capabilities are going to be integrated into the most diffused development platforms and operating systems. This trend has provoked the production of middlewares for industrial automation, home automation, multimedia, etc. In the paper a solution to model a Multimedia middleware is proposed as related to AXMEDIS EC R&D Project in which an MPEG21 compliant content factory is modeled. In that case, the middleware is used for sharing software components dedicated to the content processing and production. In the paper, the architecture of the DISIT-M3W is presented, depicting the general architecture, the class structure, the interactions among components and services, etc. by using the UML formalization.

1 Introduction

The state-of-the-art of the middleware is quite articulated. The first implementations of middleware were mainly realized as interoperability layers among heterogeneous computer based systems for general purpose usage. More recently, specific middlewares have been proposed while those with general capabilities are going to be integrated into the most diffused development platforms and operating systems. This trend has provoked the production of middlewares for workflow applications, peer to peer applications, GRID, industrial automation, home automation, multimedia, etc.

Most of the proposed component-based middleware models share the same main infrastructure features, such as the (i) support for the reuse of components, (ii) portability of the framework among different platforms, (iii) interoperability of the framework by using different languages, (iv) publication of distributed services and interfaces, (v) remote method invocation and communication, (vi) dynamic binding of services, (vii) remote object instantiation, etc.

In the case of middleware for multimedia applications, the effective needs are much more complex for the presence of additional requirements that are not commonly covered by classical component-based middlewares, such as:

- Communication: security, protocol abstraction, streaming, download, etc.
- Real-time behavior, insertion/satisfaction of real-time constraints and scheduling, concurrency, etc.
- Robust and reliable behavior, the possibility to intrinsically realize fault-tolerant solutions,
- Component security, component identification and certification,
- Interoperability among different platforms
- Localization of components, discovery, negotiation of their capabilities,
- Dynamic managing/binding (replacing, installation, configuration, pooling) of services and components, publication/announcement of capabilities,
- Low resource footprint, hardware profiling and run-time assessment, resource verification and optimization, etc
- Component trading, migration of components,
- Digital rights management (DRM), for both data (digital resources) dynamic change of protection tools, attack detection, etc.;
- Persistency, Content Management System (CMS), access to the database, access to file system, etc.: managing home repositories; information retrieval;
- Error handling/reporting and configuration management.

Some of the most important middleware models are (in some cases, strictly related to multimedia functionalities): CORBA\CCM, DCOM, .NET, EJB, JINI, Robocop, PECOS, RUBUS, MPEG M3W, MythTV, PECT, PBO, UPnP, OSGI, HAVi [1], KOALA, etc.

In this paper, the DISIT MultiMedia MiddleWare, DM3W, is presented. It has been proposed in response to the MPEG call for the M3W, and thus as contribution in the MPEG M3W group. This work has been partially developed for the integrated project

AXMEIDS IST FP6 Research & Development of the European Commission [2].

2 Scenarios

In this section, two main Multimedia middleware scenarios of our interest are reported (they are related to the AXMEDIS project [2]). They are examples in which multimedia features should be usable in a transparent manner for the end-users irrespective of their location: the home network of multimedia devices and the content factory.

2.1 Home network of multimedia devices

In this scenario, a user in his/her home environment could use transparently any of the multimedia components/services for which the grant is provided in legal and digital licensing terms (see MPEG-21 or other means) at disposal in the DM3W-enabled devices. Note that the PCs have the most general and flexible profile. Thus, a PC may act as component dispatcher or service provider for the several devices that may be present. Even some devices that do not host the DM3W RunTime can use the components registered to any DM3W RunTime they can reach. Component vendors can simply reuse the DM3W RunTime features to deliver software components to the user and the latter can treat his/her own software as a resource to be archived in a storage device for further/different usages, according to a contract/license if any. Applications can be developed and/or reused on different platforms; the life time of these platforms is guaranteed by the DM3W RunTime features (i.e., simply requesting new services to the DM3W RunTime, which provides retrieving and installation).

2.2 Processing of multimedia resources in a content factory

In this scenario, a user (i.e., any human or automatic operator in a content factory) can use his/her workstation to cooperate with several services located in the content factory on many different specialized servers for content processing. The latter use the onboard DM3W RunTime in order to execute service requests coming from the multimedia processing application installed on the workstations. The services can be of coding, decoding, adaptation, processing in general such as tools for: composition and formatting, content descriptor extractor, estimation and enforcing watermark, estimation of fingerprint, protection, packaging, transcoding, and many many others. Please note that, the management of components, which are located in the workstations, can exploit the management services of the DM3W RunTime in order to update their software components taking them from other updated sources and not mandatory from the component server, if the grant in terms of license allows this. In alternative, the simple exploitation of a remote service is a different solution to exploit the needed services and the new components located in other workstations. In the former case, the obtained component is received from the DM3W RunTime of an updated workstation, as soon as it is required from the local application. Please note that, a potential network outside the content factory could be realized by using DM3W RunTime features and capabilities. Thus partnership among content producers, components and service providers (remote execution) can be set up on the basis of the licensing mechanisms, authentication of the companies hosting DM3W RunTimes and thus performing e-commerce of components as products or establishing maintenance or renting services (pay per service, pay per byte of processing, pay per time of usage, etc..).

3 General architecture

The DM3W architecture presents a component model comprised of three entities: the manifest, the executable unit and the documentation.

The **manifest** is a metadata section describing the component. The manifest consists on an XML machine-readable information including identifiers of the components, offered services, description of their interfaces, a list of dependencies for each services (for example, a service may depend on several other services for its correct execution), and all the relevant information needed by the system to manage the lifecycle of components and services.

The **executable unit** contains the executable code of the component. The component has to be realized according to an appropriate interface defined by the DM3W architecture reported in the sequel.

The **documentation** is the human-readable compendium introduced since a component can be sold for integration or packaging purposes.

The architectural model created distinguishes between three main entities: Applications, DM3W RunTime and Components. It is expected that Applications use

DM3W RunTime to request execution of services and Components. The DM3W RunTime takes care of the execution using the components and returning the results to the *Requesters* (that in this paper are defined as Applications, Components or other RunTimes). The interaction model among them is based upon interfaces, as depicted in Fig. 1 in which the interactions are shown.

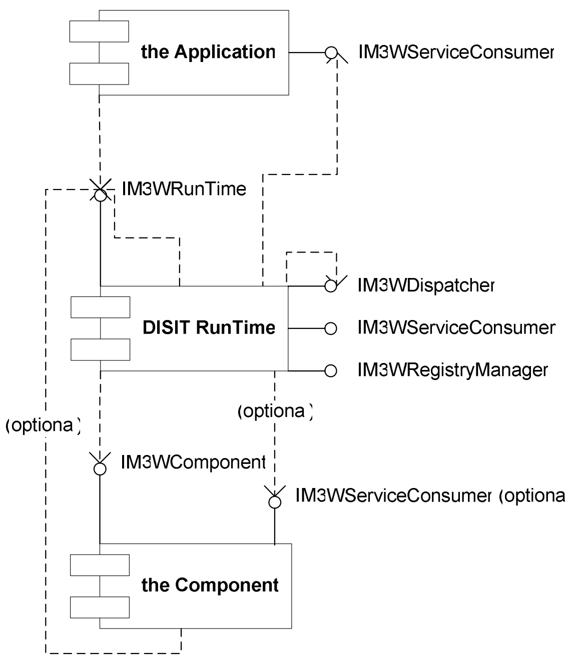


Fig. 1 – Relations between elements

The DM3W RunTime expects that users realize the **IM3WServiceConsumer** interface, which defines the bare minimum set of methods, such as callback function and remote communication methods. The mechanism to obtain service execution is based on the *Observer* design pattern [3]. Thus, a callback method is required and the users must subscribe to a notification list to get the result of the execution.

Requesters exploit the **IM3WRunTime** interface realized by the DM3W RunTime to interact with it. This interface exposes the methods needed to request the execution of a service, those that implements the Observer pattern, and others for obtaining the description of a service interface to check the syntactical accuracy of requests to be made, etc.

Please note that, all the requests are managed by the DM3W RunTime in a consistent way, through this same interface, with no regard for the kind of the Requester. That could be another DM3W RunTime, an Application or a Component.

In the proposed DM3W architecture, the RunTime of a device can contact other RunTimes (on different devices) and forward to them requests to be satisfied, thus acting as a Requester for the other RunTimes. As shown in Fig. 1, every DM3W RunTime realizes **IM3WDispatcher** interface. This interface is only accessible for DM3W RunTimes and not publicly available to other Requesters. When a service is not available on the local storage device, the RunTime which needs it can request that to other RunTimes.

DM3W compliant components have to realize the **IM3WComponent** interface to allow the DM3W RunTime to manage them. This interface defines the three methods used to acquire the use of a component, execute a service and release the component. Since a service may require the execution of other services (called *dependencies* of the former), a component can realize the IM3WServiceConsumer interface, while this is not required.

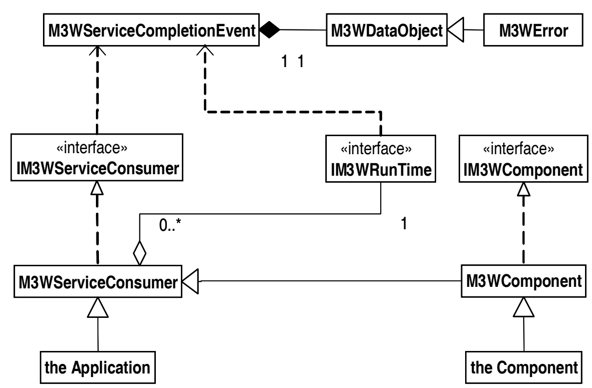


Fig. 2 - Class diagram of development-support classes

In order to simplify the development process of applications and components, a complete set of support classes have been conceived. This is shown in Fig. 2, where it can be seen that applications (and in general all Requesters of the DM3W RunTime) inherit from *M3WServiceConsumer*, which in turn realizes IM3WServiceConsumer interface needed to interact with the RunTime.

Components can be easily created specializing the *M3WComponent* class, which realizes the IM3WComponent interface used by the RunTime to manage components.

4 Middleware mechanisms and interactions

The DM3W RunTime main feature is to provide service execution. This is typically performed by means of their Identifier. To accomplish this goal, the RunTime maintains a registry of associations between components and services. This list is automatically filled up by the system at start up through an automatic exploration of all the components found in the device. During the exploration, the RunTime loads in memory the manifest of each component. The manifests are used to automatically discover which services are offered and to update the registry with other relevant information. In this phase, a verification of needed resources against available can be performed [4], [5], [6]. To this end, temporal logics and other solutions can be used [7], [8].

The IM3WRegistryManager interface can be accessed by administrative applications to trigger a new exploration phase as desired.

When the registry is set up, the DM3W RunTime is ready to serve requests. All the requests are managed by the DM3W RunTime in an asynchronous mode. In fact, RunTime has to be ready to serve others requests as soon as it has started the execution of a service. In fact a launched service may request the execution of other services and this process has to be managed in asynchronous manner in order to avoid the generation of deadlock locking conditions in which one service can try to active another that is waiting directly or indirectly for itself.

The same asynchronous mechanism has to be used in the interaction between Requesters and RunTime. In fact, the method invoked by the Requesters on a proxy of the DM3W RunTime that is active on the same device cannot return the result of an execution, because it is not able to know when the execution will finish.

Therefore, an event driven mechanism has been used, not only between the requesters and the DM3W RunTime but also between RunTime and components.

5 Conclusions

The recent trend has led to the production of specific middlewares for industrial automation, home automation, multimedia, etc. In this paper, a solution to model, design and implement a multimedia middleware has been proposed. The solution has been developed on the basis of the MPEG M3W call for technology and to satisfy the needs of AXMEDIS IST FP6 Research & Development of the European Commission [2]. In AXMEDIS an MPEG-21 compliant content factory is created for content production, protection and distribution. In that case, the middleware is used for sharing software components dedicated to the content processing and production. The DISIT MultiMedia MiddleWare, DM3W, has been proposed in response to the above mentioned MPEG call for the M3W. Thus, it has been accepted as valid contribution in the MPEG M3W group for the definition of a component model which allows remote utilization of services and components. The solution provided is mainly based on the control and execution of services that in turn may instantiate and manage remote objects. Thus the communication among distributed objects is still possible but managed as in terms of independent services. The future work will be focused in creating and integrating a resource model and behavior verification.

6 Acknowledgements

The authors would like to thanks to all AXMEDIS project partners including the Expert User Group and all affiliated members, for their contributions, funding and collaborations. A specific acknowledgment to EC IST FP6 for the partial funding of AXMEDIS project.

7 References

[1] R. Baler, C. Gran, A. Scheller, A. Zisowsky, "Multimedia Middleware for the Future Home".

[2] AXMEDIS: www.axmedis.org

[3] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. "Design Patters: Elements of Reusable Object oriented Software". Addison Wesley, 1995.

[4] Crnkovic and M. Larsson. "Building Reliable Component-Based Software Systems". Artech, 2002.

[5] J. Muskens and M. Chaudron. "Integrity management in component based systems". In Proceedings of the 30th EUROMICRO conference, Rennes France, Aug. 2004.

[6] K. Sandstrom, J. Fredriksson, and M. Akerholm. "Introducing a component technology for safety critical embedded real-time systems". In 7th ICSE Workshop on Component-Based Software Engineering, May 2004.

[7] P. Bellini, A. Giotti, P. Nesi, "Execution of TILCO temporal logic specifications" Proceedings of Eighth IEEE Intern. Conf. on Engineering of Complex Computer Systems, ICECCS'02, pp.78 -87, 2002

[8] P. Bellini, M. A. Bruno, P. Nesi, "Verification of External Specifications of Reactive Systems", IEEE Transactions on Systems Man and Cybernetics, IEEE Press, USA, pp.692-709, Vol.30, n.6, Nov 2000.

Cost Estimation Modeling Techniques for Web Applications: An Empirical Study

G. COSTAGLIOLA, S. DI MARTINO, F. FERRUCCI, C. GRAVINO, G. TORTORA, G. VITIELLO

Dipartimento di Matematica e Informatica, Università degli Studi di Salerno Via Ponte Don Melillo, 84084 Fisciano (SA), Italy

E-mail:{gcostagliola,sdimartino,fferrucci,gravino,tortora,gvitiello}@unisa.it

Abstract

The problem of estimating the effort required to develop web applications represents an emerging issue in the field of web engineering. In the paper, we report on an empirical analysis we have carried out in order to construct suitable prediction models. In particular, we have considered several features characterizing web applications and constructed some prediction models by employing different techniques, such as multiple linear regression, stepwise regression, and regression tree. The empirical analysis has been performed by using two different data sets: the first was obtained by considering web projects developed by academic students while the second is related to web projects developed by a software company. Indeed, the analysis had a twofold goal. On one hand, it allowed us to establish which features could be considered indicators of effort development and which technique could be suitable to construct a prediction model. On the other hand, it allowed us to analyze possible differences/similarity in the empirical results obtained with the two different data sets.

1. Introduction

Current web applications are often rich multimedia applications, characterized by a significant amount of images, movie clips, and audio clips. Moreover, in the last years, the web has become not only a mechanism for sharing information but also a way to access services. In few years we have passed from traditional web sites providing navigation mechanisms to sophisticated and complex web applications characterized by functionality that affects the state of the business logics. Thus, the complexity and size of web applications have dramatically augmented and there is the need for tools supporting project development planning with reliable cost and effort estimations. Indeed, effort estimation is crucial for the management of software projects since it allows to guarantee competitive bids and to effectively handle resources during the whole development process.

In the context of traditional software engineering, many software measures have been defined to gather information about relevant aspects of software products and then manage their development. In particular, several size measures have been conceived to be employed in effort/cost models to predict the effort and cost needed to design and implement the software [6]. A lot of empirical studies have been carried out to assess the effectiveness of the proposed metrics. In particular, often empirical hypothesis have been verified against observations obtained during controlled experiments [1]. On the contrary, in the literature we can find few examples of effort prediction models for web applications and few empirical case studies have been carried out to analyze metrics for web application development effort. In this context, it is worth to mention the work by Mendes *et al.* who proposed some size measures for hypermedia applications, which involve the use of variables/predictors obtained by analyzing web components [4][6]. In particular, they employed several modeling techniques, such as *case-based reasoning*, *linear regression*, and *regression trees*, that have been widely adopted for cost estimation in the context of empirical software engineering [2][9][10].

It is widely recognized that to test empirical hypotheses and/or to confirm empirical results, or to identify common trends and differences, or to assess the goodness of prediction models, several empirical studies should be performed, also by replicating experiments [1][2]. Taking into account such consideration, we have performed an empirical analysis in order to identify some features characterizing web applications (e.g., number of pages, number of medias, number of client and server side scripts) that could be considered indicators of effort development. For the analysis, we have employed different techniques, such as linear regression and regression tree, in order to determine which one could be the most appropriate. Data exploited in our analysis came from students' projects and industrial world. As a matter of fact, we have conducted a controlled experiment with undergraduate students of advanced courses on Software

Engineering and collected information on 44 web projects. Moreover, we have collected data from 15 web projects developed by a software company. The aim of this double analysis was to analyze possible differences/similarity in order to establish the reliability of the empirical results obtained from the student data set. Indeed, this is a relevant issue in the context of empirical software engineering.

The paper is organized as follows. In Section 2 the data sets used for the empirical evaluation, the employed modeling techniques, and the evaluation criteria are described. Section 3 presents the application and the comparison of the proposed development effort estimation techniques. Final remarks conclude the paper.

2. The method

In this section we first present the data sets used in the empirical analysis, and then we briefly describe the modeling techniques employed and the criteria adopted to assess the acceptability of the derived effort prediction models.

2.1 The data sets

The first data set has been obtained by exploiting data coming from 44 web projects developed by undergraduate students of courses on Software Engineering. Students were organized in groups including one project manager, two web designers, two web programmers and one tester. The project manager was a student of an advanced academic course on Software Engineering. To allow uniformity among the groups the most skilful students were equally distributed. Each group was asked to implement a web application and to record information on the actual effort required for the development process in terms of person-hours. They designed and developed web applications in different application domains, such as e-commerce, e-learning, and web portals.

We have worked to make the empirical evaluation and the controlled experiment as accurate as possible in order to avoid that the involvement of students could bias the external validity of the empirical results. As a matter of fact, the subjects involved in our experiment were final-year undergraduate students, who can be considered young professional developers and programmers, or in any case they had skills similar to professionals which start their web development career. The analysis of the documentations and the final implementations showed that the dimensions of web applications developed by the students can be compared to those developed by professionals. Furthermore, we have carefully considered any other factors that could bias the results of an empirical study [3][6]. In particular,

- to mitigate the effect of instrumentation we have used the same questionnaire to collect information,
- to overcome the difficulties of measuring web applications we have applied systematic rules on artifacts, and
- to reduce learning effects students designed and built a web application as course-work before starting the design and the development of applications for the experiment.

The second data set has been obtained by exploiting data coming from 15 web projects developed by a software company. The analyzed web applications fall in the following domains: e-government, cultural heritage, e-commerce, and web portals.

Table 1 contains the variables we have considered in our analysis in order to collect information on the projects that can be exploited to construct and evaluate effort prediction models. They represent early size metrics that can be obtained from analysis and design documents.

Table 1. Variables for the web applications[1]

Variable	Description	scale
EFH	Effort in person hours to develop the web application	Ratio
Wpa	Number of web pages	Ratio
N_Wpa	Number of new web pages	Ratio
Me	Number of multimedia elements	Ratio
N_Me	Number of new multimedia elements	Ratio
R_Me	Number of reused multimedia elements	Ratio
CSA	Number of Client side Scripts and Applications	Ratio
SSA	Number of Server side Scripts and Application	Ratio
IL	Number of Internal links	Ratio
EL	Number of External references	Ratio

2.2 The Modeling Techniques

We have used the linear regression analysis which is one of the most commonly used statistical techniques for exploring the relationship between a dependent variable and one or more independent variables [8]. In particular, the relationship is described by an equation $y = b_1x_1 + b_2x_2 + ... + b_nx_n + c$, where y is the dependent variable; x_1, x_2, ..., x_n are the independent variables; for i=1,…,n, b_i is the coefficient that represents the amount the variable y changes when the variables x changes 1 unit; c is the intercept.

Several crucial indicators have been taken into account to evaluate the quality of the resulting models. In particular, we have determined the goodness of fit of a regression model by using the square of the linear correlation coefficient, R^2, which indicates the amount of the variance of the dependent variable that is explained by

[1] Variables *Wpa*, *N_Wpa*, *Me*, *N_Me*, and *R_Me* have been used also by Mendes *et al*. [5][6].

the model related to the independent variable. Furthermore, the *F* value and the corresponding *p-value* (denoted by *Sign F*) are useful indicators of the degree of confidence of the prediction. Specifically, a high *F* value and a low *p-value* indicate that the prediction is indeed possible with a high degree of confidence. We have also considered the *p-values* and *t-values* for the corresponding coefficients and the intercept. The *p-values* give an insight into the accuracy of the coefficients and the intercept, whereas their *t-values* allow us to evaluate their importance for the generated model. In particular, *p-values* less than 0.05 are considered an acceptable threshold, meaning that the variables are significant predictors with a confidence of 5%. As for the *t-value*, a variable is significant if the corresponding *t-value* is greater than 1.5, which is the case for the coefficients and the intercept.

We have also applied a forward stepwise regression, which allows us to compute linear regression in stages [8]. In the first stage, the independent variable that is best correlated with the independent variable is considered in the equation. Then, in the second stage the analysis focuses on the remaining independent variables, and the one characterized by the highest correlation with the dependent variable is included in the equation, by taking into account the first independent entered. The analysis continues until the addition of a remaining independent variable do not improve R^2 by a significant amount.

Regression tree is a variant of decision trees that can be used to approximate real-valued functions[1][2]. It takes in input a set of numerical variables and generates a regression tree to predict the value of the target variable that is analogous to the dependent variable in linear regression. A binary tree is built by recursively splitting the input data into partitions. At the beginning all data are associated to the root, and they are split into two parts so that it minimizes the sum of the squared deviations from the mean in the separated parts. At each split the process determines which input variable has to be used for splitting, which set of input values of the variable is associated to the left child node, and which set is associated to the right child node. Observe that each node has associated the mean value for the target variable. The process ends when for each node a minimum size specified for the node by the user is obtained.

Thus, to determine the predicted value for the target variable, we start from the root node and then decide to follow right or left child based on the value of the splitting variable. We continue until a leaf node is reached. The mean value for the target variable specified in this node is the predicted value.

As suggested in [2] we also combined the use of regression tree and linear regression. In particular, we have first developed a regression tree, and then we have applied linear regression to the projects contained in each terminal node. The regression equations obtained for each terminal node is used to predict the development effort for the projects in the node.

The above modeling techniques have allowed us to construct cost estimation models that can be used to predict the development effort in terms of the number of person-hours required. In order to have more reliable non-biased results, we have followed a multiple-fold cross validation, partitioning each data set into two randomly selected sets: the training set for model building and the test set for model evaluation. Indeed, when the accuracy of the model is computed using the same data set used to build the prediction model, the accuracy evaluation is considered optimistic. Furthermore, cross validation is widely used to validate cost estimation model when dealing with small data sets (see, e.g.[2]).

2.3 Evaluation Criteria

In order to assess the acceptability of the derived effort prediction models, we have considered the *Magnitude of Relative Error*, which is defined as

$$MRE=|EFH_{real}—EFH_{pred}\,/EFH_{real},$$

where EFH_{real} and EFH_{pred} are the actual and the predicted efforts, respectively. In particular, for each test set we have evaluated the prediction accuracy by taking into account a summary measure, given by the *Mean of MRE* (*MMRE*), to measure the aggregation of *MRE* over the observations contained into the test set. As suggested by Conte *et al.* in [3], an acceptable threshold for an effort prediction model is given by an *MMRE* value less than 0.25.

Moreover, we have considered another meaningful measure, namely the *prediction at level l*, defined as

$$PRED\ (l)= k\ /N$$

where *k* is the number of observations whose *MRE* is less than or equal to *l*, and *N* is the total number of observations. Again, according to Conte *et al.*, a good effort prediction model should have $PRED(0.25) \geq 0.75$.

3. Application and Comparison of development effort estimation techniques

Table 2 contains the descriptive statistics performed for the variables of Table 1 by considering the 44 projects developed by the students and the 15 projects developed by the software company.

The application of a forward stepwise regression analysis on the data set obtained from the students' projects has given rise to the four models depicted in Fig. 3.1(a). In particular, we have applied a 4-fold cross validation approach by constructing 4 testing set formed by 11 observations. All the models uses *SSA* as independent, which results to be the best correlated with the dependent variable *EFH.* We can observe that the linear regression analysis shows a quite high R^2 value for all the four models. As an example, let us consider the training set no.1. An R^2=0.671 indicates that 67.1% is the amount of the variance of the dependent variable *EFH* that is explained by the model related to *SSA.* The goodness of the prediction models is confirmed by *F* values which are high and by *SignF* values which are 0.000. Furthermore, *p-values* and *t-values* for the independent variables and the intercept are also less than 0.05.

Table 2. Descriptive statistics of EFH and variables collected in Table 1

	Obs	MIN	MAX	MEAN	STD. DEV.
Students' Projects					
EFH	44	62	183	126,796	33,506
Wpa	44	2	31	10,750	7,409
N_Wpa	44	2	27	10,023	6,330
Me	44	2	54	17,727	13,904
N_Me	44	2	49	14,500	11,170
R_Me	44	0	13	3,227	3,857
CSA	44	1	198	27,227	34,908
SSA	44	14	179	79,818	45,955
EL	44	0	15	3,955	4,918
IL	44	7	87	29,477	21,204
Software Company Projects					
EFH	15	1176,00	3712,00	2677,867	827,115
Wpa	15	2,00	46,00	17,000	12,317
N_Wpa	15	2,00	46,00	17,200	12,167
Me	15	54,00	223,00	104,133	43,533
N_Me	15	20,00	223,00	82,533	56,599
R_Me	15	,00	51,00	21,600	23,999
CSA	15	5,00	55,00	26,933	16,918
SSA	15	21,00	209,00	80,400	55,414
EL	15	,00	8,00	4,933	3,770

For the software company data set, we have applied a 3-fold cross validation approach by constructing 3 testing sets formed by 5 observations. We have obtained three prediction models which use *SSA*, *CSA*, and *IL* as independent variables and are characterized by high R^2 values (see Fig. 3.1(b)). However, the models for sets *SRa1* and *SRa3* have low *F* values which can introduce a doubt about the goodness of the models. Furthermore, the *t-value* and *p-value* characterizing the intercept are less than 1.5 and greater than 0.05 respectively.

A multiple linear regression analysis has been applied by considering the predictors that are statistically correlated to the effort (*EFH*) at level 0.05. To this aim, we have carried out both Spearman's rho and Pearson correlations.

For the student data set, Pearson correlation has revealed that variables *Wpa*, *N_Wpa*, *R_Me*, *SSA*, and *IL* are statistically correlated to the effort at level 0.05. While Spearmean's rho correlation has revealed that *Wpa*, *N_Wpa*, *Me*, *R_Me*, *SSA*, *CSA* and *IL*, are statistically correlated to the effort at level 0.05. In both cases, *SSA* is the variable characterized by the highest correlation factor, i.e., 0.821, while the others vary from 0.386 to 0.577. If we consider only *SSA* as independent variable the multiple linear regression analysis boils down to stepwise regression. However, we have also performed a linear regression analysis by considering *EFH* as dependent variable and *Wpa*, *N_Wpa*, *R_Me*, *SSA*, and *IL* as independent variables. The four prediction models obtained are characterized by quite high R^2 values, while the *F* values are not so high (see Fig. 3.2(a)). Moreover, let us observe that for the variables *Wpa*, *N_Wpa*, *R_Me* and *IL* the *p-values* are greater than 0.05 and the *t-values* are less than 1.5. This means that those variables cannot be considered significant predictors with a high degree of confidence.

	R^2	F	SignF
SRs1	0.671	63.363	0.000
SRs2	0.674	64.055	0.000
SRs3	0.666	61.942	0.000

(a)

	R^2	F	SignF
SRa1	0.903	18.653	0.002
SRa2	0.957	44.071	0.000
SRa3	0.815	8.838	0.013

(b)

Figure 3.1. The prediction models obtained by applying the stepwise regression

	R^2	F	SignF
MRs1	0.686	11.798	0.000
MRs2	0.679	11.4	0.000
MRs3	0.673	11.12	0.000
MRs4	0.730	14.05	0.000

(a)

	R^2	F	SignF
MRa1	0.552	4.309	0.060
MRa2	0.721	9.067	0.011
MRa3	0.357	1.945	0.213

(b)

Figure 3.2. The prediction models obtained by applying the multivariate regression

For the software company data set, both Spearman's rho and Pearson correlations have revealed that *SSA* and *IL* are statistically correlated to the effort at level 0.05, where *SSA* is characterized by a higher correlation factor than *IL*. By applying a 3-fold cross validation approach we have obtained the three prediction models presented in Fig. 3.2(b). Observe that the models for sets *MRa1* and *MRa3* have low R^2 values and *SignF* values greater than 0.05. Furthermore, *t-values* and *p-values* for *SSA* and *IL* suggest that these variables cannot be considered significant predictors with a high degree of confidence.

We have applied the regression tree technique by considering the variable *EFH* as target and the variables

Wpa, *N_Wpa*, *Me*, *R_Me*, *CSA*, *SSA*, *EL* and *IL* as predictors. Moreover, we have chosen the maximum splitting level and the minimum size node to split equal to 10, and a 10-fold cross validation to validate and prune the tree.

By considering the student data set, the tree size summary report has shown that the full tree has 8 terminal (leaf) nodes but the minimum validation relative error occurs with 2 nodes. Indeed, the relative error value is 0.4835 with a standard error of 0.1084, and with a standard error for tree size reduction equal to 0.500, and the optimal tree has 2 nodes. Thus, the tree has been pruned from 8 terminal nodes to 2 terminal nodes. In particular, the root of the tree is characterized by a mean value of *EFH* equal to 126.795 and a standard deviation of 33.123. The group has been split on the variable *SSA*, and two terminal nodes can be reached. The left child node, numbered 2, has 14 rows associated and a case goes left if *SSA* ≤ 47.5. The right child node, numbered 3, has 30 rows and a case goes right if *SSA* > 47.5. Furthermore, the mean value of *EFH* in node 2 is 89.143 and the standard deviation of *EFH* is 20.712, while for node 3 the mean value and the standard deviations for *EFH* are 144.367 and 20.944 respectively.

We have also applied regression tree on the data set obtained from software company's projects even though it is a small data set. The tree size summary report obtained has shown that the minimum validation relative error occurs with 2 terminal nodes. The root of the tree is characterized by a mean value of *EFH* equals to 2677.867 and a standard deviation of 799.069. The group has been split on the variable *SSA*, and two terminal nodes can be reached. The left child node 2 can be reached if *SSA* ≤ 63 and has 7 rows associated. The right child node 3 can be reached if SSA>63 and has 8 rows associated. Furthermore, the mean value of *EFH* in node 2 is 1962.286 and the standard deviation of *EFH* is 496.142. While the mean value of *EFH* in node 3 is 3304 and the standard deviation of *EFH* is 376.468.

As for combination of the regression tree and linear regression analysis, we have performed a linear regression analysis on the observations associated to the two leaf nodes of the regression trees. For both data sets we have built two prediction models which consider variable *SSA* as independent variable and *EFH* as dependent variable (see Figg. 3.3(a) and 3.3(b)).

For the student data set, the first model is built by considering the 14 observations characterized by *SSA*≤47.5 while the second is built by considering the 30 observations characterized by SSA>47.5. For the software company data set the first model is built by considering the 7 observations characterized by *SSA*≤63 while the second is built by considering the 8 observations characterized by SSA>63.

The two models obtained by considering the student data set are characterized by low R^2 values and the F values do not present values so high. However, for the variable *SSA* and the intercept, the *p-values* are less than 0.05 and the *t-values* are greater than 1.5. The two models obtained for the software company data set are characterized by very low R^2 and F values, and for the variable *SSA* and the constant the *p-values* are greater than 0.05 and the *t-values* are less than 1.5.

	R^2	**F**	**SignF**
SSA<=47.5	0.343	6.265	0.028
SSA>47.5	0.328	13.653	0.001

(a)

	R^2	**F**	**SignF**
SSA<=63	0.22	1.40	0.290
SSA>63	0.02	0.13	0.728

(b)

Figure 3.3. The prediction models obtained by combining regression tree and linear regression

In order to evaluate and compare the accuracy of the analyzed prediction models we have taken into account the *MMRE* and *PRED*(0.25). In particular for each analyzed technique, once accuracy has been separately calculated for test sets, the resulting values have been aggregated across all the sets. Table 3 reports the aggregate *MMRE* and aggregate *PRED*(0.25) values.

We can observe that all the analyzed models seem to be good for estimating the development effort. Indeed, they exhibit an aggregate *MMRE* value less than 0.25 that represents an acceptable threshold for an effort prediction model, as suggested in [3]. Moreover, all the models are characterized by an aggregate *PRED*(0.25) greater than 0.75, except that derived in the multivariate regression analysis.

The worst results have been achieved by applying the multivariate regression and using the software company data set since the obtained model has the highest *MMRE* and the lowest *PRED*(0.25). Moreover, the analysis of the models obtained by using the multivariate linear regression has suggested that the used variables cannot be considered significant predictors with a high degree of confidence.

The best results have been achieved by combining the regression tree and the linear regression analysis, which gives rise to the lowest *MMRE* and the highest *PRED*(0.25). However, as described above, the models obtained by combining these two techniques are also characterized by low R^2 and F values. Furthermore, for the software company data set also *p-values* and *t-values* for variables and intercept do not present reasonable results, which introduce serious doubts about the goodness of fit of the regression models and the significance of the predictors.

Let us observe that the application of the only regression tree has also achieved interesting results, which are very closed to those obtained with stepwise regression. Indeed, for the student data set there is no statistical significance difference in the *MMRE*s between these two techniques. On the contrary, for the software company data set stepwise regression presents better results with respect to the regression tree. However, the software company data set is small and the regression tree analysis requires a larger data set.

Table 3. Aggregate accuracy evaluation

	Student Data Set		Softw Company Data Set	
	MMRE	*PRED(0.25)*	*MMRE*	*PRED(0.25)*
MR	0.20	0.68	0,23	0,67
SR	0.15	0.80	0,09	1
RT	0.16	0.82	0.17	0.8
RT+LR	0.13	0.90	0.14	0.86

MR = multiple regression, SR = stepwise regression, RT = regression tree, RT+LR = combination of regression tree and linear regression

It is worth pointing out that these results slightly differ from those provided by Mendes *et al*. [4], where the tests of significance for *MMRE*s and *PRED*(0.25) indicated that linear regression models gave statistically significant better results than regression tree models. Our analysis confirms the usefulness of stepwise regression, but further investigation is needed for regression tree analysis (also with larger data sets).

Moreover, observe that the models obtained by using regression tree and stepwise regression are based on the use of the variable *SSA*. Thus, according to our analysis *server side script and applications* turns out to be the most significant variable to estimate the web application development effort.

4. Final Remarks

We have provided an empirical study meant to analyze some development effort estimation models for web applications. In particular, we have employed multiple linear regression, stepwise regression and regression tree by using predictors such as number of pages, number of medias, number or reused medias, number of client and server side scripts. The analysis has been carried out by using data collected from two sets of projects. The first set is formed by projects developed by students of advanced courses on Software Engineering. While the second set consists of projects developed by a software company operating in our territory. The study has shown that there is not significant difference in the results obtained from the two sets of projects. In particular, in both cases the best results are achieved by employing stepwise regression and regression tree, while multivariate regression analysis does not provide interesting results. This provides evidence that the use of student projects can be useful to carry out empirical analysis when accurate controlled experiments are performed and actions are taken in order to ensure the external validity of the empirical results. This can be an important indication for empirical software engineers because since collecting data coming from the industrial world is a hard task. However, in the future it is necessary to further verify this observation by using a bigger set of industrial projects. Moreover, we plan to employ other techniques, such as case-based reasoning in order to analyze their ability to construct suitable prediction models for web applications [4][6][8].

References

[1] V. Basili, F. Shull, and F. Lanubile, "Building Knowledge through Families of Experiments", *IEEE Transactions on Software Engineering*, vol. 25, no. 4, pp.456-473, 1999.

[2] L. Briand, K. E. Emam, D. Surmann, I. Wiekzorek, and K. Maxwell "An Assessment and Comparison of Common Software Cost Estimation Modeling Techniques," in *Proceedings of International Conference on Software Engineering* (*ICSE'99*), Los Angeles, USA, 16-22 May, pp. 313-322, 1998.

[3] D. Conte, H.E. Dunsmore, V.Y. Shen, *Software engineering metrics and models*, The Benjamin/Cummings Publishing Company, Inc., 1986.

[4] B. A. Kitchenham, S. L. Pfleeger, L. M. Pickard, P. W. Jones, D. C. Hoaglin, K. El Emam, J. Rosenberg "*Preliminary guidelines for empirical research in software engineering"* IEEE Transactions on Software Engineering, vol. 28, no. 8, pp. 721 – 734, 2002.

[5] E. Mendes, I. Watson, C. Triggs, N. Mosley, S. Counsell, "A Comparison of development Effort Estimation Techniques for Web Hypermedia Applications", in *Proceedings of 8th International Software Metrics Symposium* (*METRICS'02*), 4-7 June, Ottawa, Canada, pp. 131-140, 2002.

[6] E. Mendes, I. Watson, C. Triggs, N. Mosley, S. Counsell, "A Comparative Study of Cost Estimation Models for Web Hypermedia Applications", *Empirical Software Engineering* vol. 8, no. 2, pp. 163-196, 2003.

[7] Measurement in Software Engineering, http://www2.umassd.edu/SWPI/ProcessBibliography/bib-measurement.html

[8] D. Montgomery, E. Peck, G. Vining, "*Introduction to Linear Regression Analysis*", John Wiley & Sons, Inc., 3rd Ed., 2001.

[9] T. Mukhopadhyay, S. Vicinanza, M. Prietula, "Examining the feasibility of a case-based reasoning model for software effort estimation", MIS Quarterly, pp. 126-171, 1992.

[10] M. Shepperd, C. Schofield, "Estimating Software project effort using analogies", *IEEE transaction on Software* Engineering vol. 23, no 12, pp 736-743, 1997.

ImagePickup: A Web Based Hybrid Image Search Engine

Abolfazl Lakdashti[1,2] and Mohammad Shahram Moin[2]
[1]Computer Group, Islamic Azad Univ. / Science and Research Campus, Tehran, Iran.
[2]Multimedia Group, Iran Telecom Research Center (ITRC),Tehran, Iran.
E-mail: {lakdashti, Moin}@itrc.ac.ir

Abstract

The web is a very large distributed digital information space. Multimedia databases and its applications are rapidly expanding; it is because meanings and semantics transferring in visual format is more efficient than only text. Therefore it is necessary more efficient and effective search methods. In this paper, an image search engine proposed called ImagePickup that aims to overcome the semantic gap. One problem in image search engines is semantic gap where the meaning that the user has in mind for an image is at a higher semantic level than the features on which the database operates. In proposed scheme we have been used three components for image retrieval include text-based image retrieval, visual content-based image retrieval (color, texture and shape) and text-visual integration based image retrieval. Experimental results show that the user's semantics retrieval can be approximated better via linking image content and text in retrieval procedure than the single method which reported in the literature.

1 Introduction

Managing the tremendous number of images in World Wide Web and relevant databases needs more efficient and fast algorithms and tools. Image similarity measurement is one of the most important aspects in a large image database for efficient search and retrieval to find the best match for a user query. Recently researchers have reported on several World Wide Web image search engine. There are two ways to index the meaning of an image. (1) It can be encoded in textual form and attached to an image that we call it text-based image retrieval (TBIR). (2) It can be directly extracted from the image data that well-known to content-based image retrieval (CBIR). TBIR is a traditional way of accessing images in a repository is by describing them using a symbolic index that often takes the form of a set of keywords. This form of indexing has been criticized because assigning keywords to a large image database is a labor intensive operation requiring rather specialized personnel, also labeling can provide at best case a partial description of the contents of an image. Second drawback of text-based image retrieval is very hard to derive information that answer this kind of queries and to encode it in a series of keywords, while there is indication that visual search and browsing is rather effective in these circumstances. CBIR is second way to determine the meaning of image is content based image retrieval. In this way significant information directly extract from image content and store in feature vector. Significant information includes color, texture, shape and etc in compress case. One problem of CBIR has always been the semantic gap. Semantic gap is a wide gap between the subjective user's interpretation of image similarity in a given context and that of the objective similarity model used by the image search engine. Therefore semantic information of the image is not utilized enough during retrieval.

A number of general-purpose image search engines have been developed. The references below are to be taken as examples of related work, not as the complete list of work in the cited area. Some of these search engines applied both text and content for retrieval. IBM QBIC [1] is one of the systems in the commercial domain. Also, systems have been developed at IBM T.J. Watson [2]. In the academic domain, MIT Photobook [4] is one of the earliest. Berkeley Blobworld [5], Columbia VisualSEEK and WebSEEK [6] and Stanford WBIIS [7] are some of the recent systems.

This paper presents the image database system ImagePickup. The ImagePickup contains a methodology for interacting with images using both text and visual content features. The system creates two indices for each image: a lexical index, based on the words associated to the images, and a visual index, based on the visual features of the image.

The paper is organized as follows. Section 2 describes the ImagePickup framework includes: textual search engine, visual content search engine and image visual-text integration retrieval.Section 3 presents a query example . Also experimental results and conclusion will be described in section 4, 5 respectively.

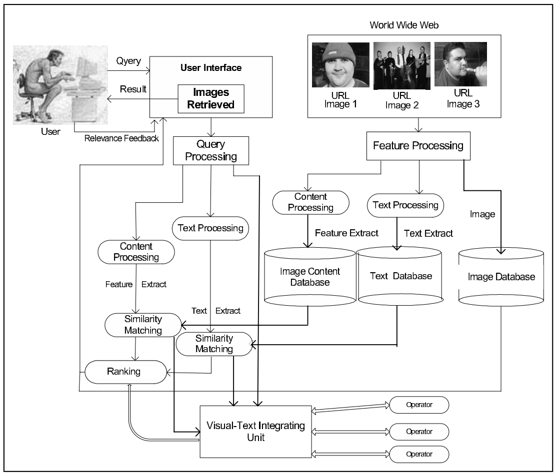


Figure 1. ImagePickup architecture

2 Framework

2.1 Architecture

In this section we describe ImagePickup architecture and structure. Figure 1 shows ImagePickup architecture. The user interface typically consists of a query formulation part and a result presentation part. Specification of which images to retrieve from the database can be done in three ways. One way is to specify the image in terms of keywords by text processing unit. Another way is to specify the image features that are extracted from the image content, such as a color, texture and shape by content processing. Third way is retrieval based on linking images and keywords for good precision retrieval, which handles by visual-text integrated unit. In our algorithm we have used a hybrid approach based on both TBIR and CBIR. Vector model approach is used for the TBIR subsystem (text processing). We have used color histogram model to extract color features. This model is fast for large image databases. For image texture feature extraction we used both co-occurrence matrix and complex wavelet transform with dual tree (DT-CWT). Our experiments showed that DT-CWT performs better than the co-occurrence matrix. We have also used descriptor elements such as area, centroid and etc for shape representation. Finally, we have used a linear combination of these features with weighted coefficients to implement our CBIR system.

2.2 Text-based image retrieval

Textual search is done using a vector space approach [2]. The set of labels associated to each image is considered as a vector in a high dimensional space having one axis per keyword, and the similarity between two vectors is computed as the cosine of the angle between them. In spite of the prohibitively high dimensionality of this feature space, the computation of the distance is rather efficient, since there is never the need to consider all the dimensions of the space at once.

$$\left[\left\{\begin{array}{l} W(I) = (w_{1I}, w_{2I}, \cdots, w_{tI}) \\ W(J) = (w_{1J}, w_{2J}, \cdots, w_{tJ}) \end{array}\right.\right] \tag{1}$$

If *W(I)* is the set of keywords associated to the image *I*, and If *W(J)* is the set of keywords associated to the image *J* then the cosine of the angle $\theta_{I,J}$ between image *I* and Image *J* can be computed as:

$$\cos\theta_{(I,J)} = \frac{\sum_{i=1}^{t} w_{iI} \times w_{iJ}}{\sqrt{\sum_{i=1}^{t} (w_{iI} \times w_{iJ})^2}} \tag{2}$$

This angle is similarity between images *I* and *J* based on keywords associated to this images. The angle value is interval 0 and 1. Whatever angle value near to 0 similarities is more.

2.3 Visual content-based feature extraction

Figure 2 shows flowchart of the proposed method in visual content extraction that includes three successive steps. First, an image is split into two regions. We call them as the inner and outer regions with the size of 4/5 and 1/5 of the whole image size, respectively. Second, the color feature of the image content is extracted for the inner and outer parts of the image. The color-based proposed scheme will be described in section 3. Third, the texture features including the inner and outer parts of the image are extracted. Section 4 will analyze the texture-based proposed method.

2.3.1 Color feature extraction

For color feature extraction we transformed color space to HSV color space because HSV color space includes attributes: Completeness, Naturalness, Uniformity and compactness. Usually, the color histogram, color moment, color coherence vector [8] and color Correlogram [9] are used for color representation. Color Correlogram is more performance than another models, but in online applications with large scale of images, binary color set have good response time than color Correlogram. Therefore we implemented a color histogram method that almost like binary color set. The color components of the proposed descriptor are calculated for inner and outer regions, separately. We have transformed the (R, G, B) color space into a perceptually uniform space HSV, For inner region; and then quantized the transformed color space into 162-bin color histogram. These values are achieved by a uniform quantization, which includes 18 levels in H, three levels in S, and three levels

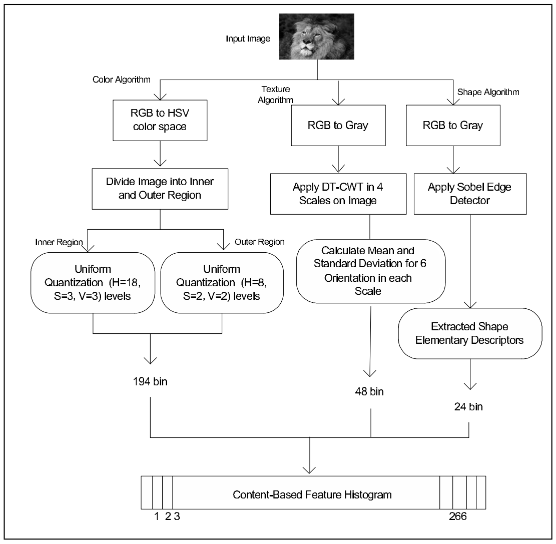


Figure 2. Visual content based feature extractor structure

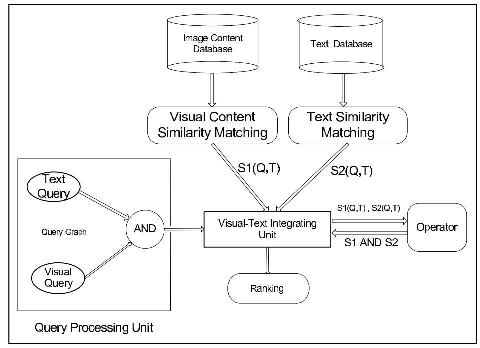


Figure 3. Visual-Text integrated unit

in V color space. The color histogram for outer region includes 32-bin achieved by a uniform quantization of HSV space with 8 levels in H, two levels in S, and two levels in V color space. The color-based proposed method uses two weighted histograms to overcome the color similarity between their bins. This model is also very fast computationally, for large image database.

2.3.2 Texture feature extraction

The texture component of the descriptor includes a 48-bin texture histogram based on complex wavelet transform coefficients with dual tree (DT-CWT). We measure the mean and standard deviations of DT-CWT transform coefficients in 24 subbands to calculate the texture features. These 24 subbands coefficients are obtained from 4 scales and 6 orientations in each scale. Therefore, resulting to 48 output features for each image.

2.3.3 Complex wavelet transform

The complex wavelet transform uses a dual tree of wavelet filters to find the real and imaginary parts of the complex wavelet coefficients [10]. Approximate shift invariance, good directional selectivity with six oriented, and computational efficiency properties of DT-CWT make it a good candidate for representing the texture features. In two dimensional, CWT basis functions modulate complex exponentials of the form

$$h(x, y) = a(x, y)e^{j(w_x x + w_y y)} \tag{3}$$

With a(x, y) a slowly varying Gaussian-like real window function centered on (0, 0) and (w_x, w_y) the center frequency of the corresponding subband. The real and imaginary parts of the CWT basis functions are symmetric and antisymmetric respectively about (0, 0); hence they are orthogonal. This means that the real and imaginary parts of each CWT coefficient are approximately statistically uncorrelated [4]. The orientation ± 15 degree, ± 45 degree, ± 75 degree is obtained since complex filters can separate positive and negative frequencies in both horizontal and vertical directions [11]. The responses of complex wavelets frequency are shown in Fig. 3. There are 6 orientations at each of 4 scales (any number of scales can be used, but the number of orientations is built into the method).

2.3.4 Shape feature extraction

The shape features consist of shape area, centroid, circularity, eccentricity, major axis orientation and a set of algebraic moment invariants. The major axis orientation and the eccentricity are computed from the second order covariance matrix of the boundary pixels. The major axis orientation as the direction of the largest eigen vector and eccentricity as the ratio of the smallest eigen value to the largest one. For the database images, these shape features are extracted for all the object contours.

2.3.5 Similarity distance

We have used a linear combination of the features with weighted coefficients in the proposed scheme as follows:

$$FeatureVector = \alpha x + \beta y + \gamma z + \eta w \tag{4}$$

In equation (4) *x, y, z* and *w* are color (inner part), color (outer part), texture feature vectors and shape feature vectors, respectively; extracted from our proposed algorithm. The values of $\alpha, \beta, \gamma, \eta$ are feature significant coefficients based on retrieval operation. In this paper we have used the

Euclidean distance metric to compare two images, one with feature vector hq, and one with feature vector ht as equation 5.

$$labeleq5Euc.d^r_{q,t} = \sqrt{\Big[\sum_{m=0}^{M-1} \big|h_q(m) - h_t(m)\big|^2\Big]} \quad (5)$$

2.4 Image Visual-Text Integration retrieval

The Visual-Text Integration (VTI) units in ImagePickup handles three process include: 1. Image and text query reception from user interface and transfer to feature processing unit (Query Reception). 2. Query dispatching from similarity unit to integration operators unit (similarity transfer). 3. Similarity process using operators, merge results and transfer to ranking section for represent of image in user interface (similarity matching). The architecture of VTI unit is schematically shown in Fig. 3.

2.4.1 Query Reception

In ImagePickup for each query makes a graph we call query graph. Query graph contains two types of nodes: Query type (for example text or visual content) and operators. Query type nodes specify that which processing unit digests query and algebra operator nodes describe how the results of queries should be combined.

2.4.2 Similarity Transfer

The VTI unit primarily computes the similarity between two images. This process is illustrated in Figure 2. Consider a query graph in Figure 3, for instance, the similarity criterion include text and color similarity, and the results of the two criteria should be joined together using the AND operator. Query color includes color saturation of desired image and text includes number of keywords to describe as the semantic of image. While users desire an image for search in image database, for instance consider two images Q and T, the text process unit returns $S_1(Q,T)$ and the content process unit returns $S_2(Q,T)$. As specified by the query graph, these two score must be joined together using the AND function. For each image, given the values $S_1(Q,T)$ and $S_2(Q,T)$ the AND operator returns the value $S(Q,T) = S_1(Q,T) \wedge S_2(Q,T)$, which is returned as the similarity between image and image with respect to the current similarity criterion [12].

2.4.3 Similarity matching

The VTI unit includes a number of algebraic operators to put together the results of the content and text processing

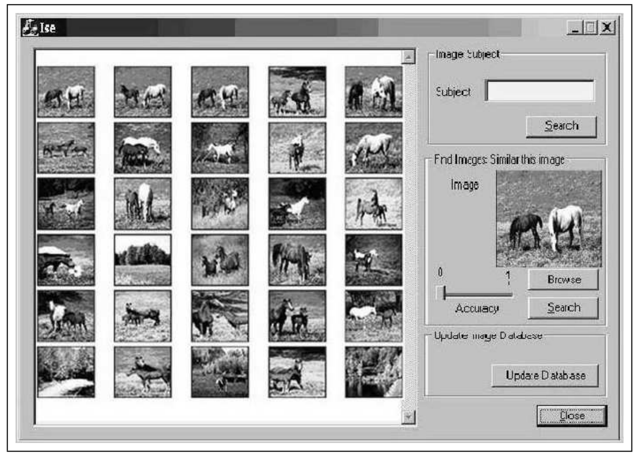


Figure 4. Interface and result of a query example

unit. The operators take two similarity measures relative to two processing unit, and transform them into a new similarity measure resulting from their combination. Consider a query Q, and let $S_{c1}(Q,T)\epsilon[0,1)$ be the similarity between the query Q and image T according to the criterion c1 implemented by content processing unit. We can interpret this value as the truth value of the predicate "Image *I* is like the query *Q* according to criterion visual content." Similarly we have the value $S_{c2}(Q,T)$ representing the similarity between Q and I according to criterion c_2. We make the following hypothesis. The similarity of *Q* and *I* image with respect to criterion "c_1 and c_2" depends only on the values $S_1(Q,T)$, $S_2(Q,T)$. In this case, we can write $S_{c1\wedge c2}(Q,T) = S_{c1}(Q,T) \wedge S_{c2}(Q,T)$. We can consider similarly algebra function $\vee$ for OR connective and for $\neg$ NOT (negation) operator. The above operator definition that makes it possible to satisfy all three relations are

$$d_1 \wedge d_2 = max(d_1, d_2) \quad (6)$$

$$d_1 \vee d_2 = min(d_1, d_2) \quad (7)$$

$$\neg d = 1 - d \quad (8)$$

3 A query example

Here we present a brief query example of use of ImagePickup and its interface. Figure 4 shows the interface of ImagePickup. The user was asked to look for a certain group of Horse images, which look like the image in right of interface (figure 4).also figure 4 shows the results after a few interactions.

4 Experimental results

We implemented the ImagePickup search engine by Java and Java server page from Sun Microsystems software programming technology and Oracle database for repository management of images and feature vector. We used a machine by CPU 1.7 megahertz Intel processor with 512 megabyte RAM. In the following we present our experimental results that achieved in simulation and construction of ImagePickup system.

4.1 Data set and performance measure

Our image data set from the Corel images collection, including 3117 images (N=3117 JPEG images) divided into 22 classes of different scenes (e.g. flowers, lions, sea, etc.). The performance of the retrieval results is measured by precision and recall metrics as follows:

$$Recall = \frac{\text{No. of relevant images retrieved}}{\text{Total no. of relevant images in database}} \quad (9)$$

$$Precision = \frac{\text{No. of relevant images retrieved}}{\text{Total no. of images retrieved}} \quad (10)$$

Two above measurements are well-known method in image retrieval systems.

4.2 Text-based retrieval

We simulated two models of text based information retrieval, Boolean model and vector model. In Boolean model we unable to ranking in image retrieval results, but in vector model we have these advantages: 1. simplicity, 2. fast search of records, 3. weighting to keywords (queries) and ranking based record retrieval. Therefore we implemented vector model for Text-based retrieval unit in ImagePickup system.

4.3 Visual content-based retrieval

In visual content-based we used three features (color, texture and shape) for representation of image content. Some experimental results are shown with color proposed method in figure 5, 6. Also we compared texture proposed method with co-occurrence matrix. Co-occurrence matrix is well known model of texture representation that proposed by Haralick. Experimental result comparisons of our method and co-occurrence matrix are shown in figure 7, 8.

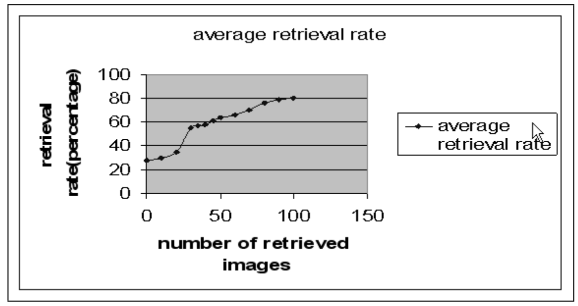


Figure 5. Retrieval rate vs. number of retrieved images (6 queries of Ducks and Trees) with color propose method

.

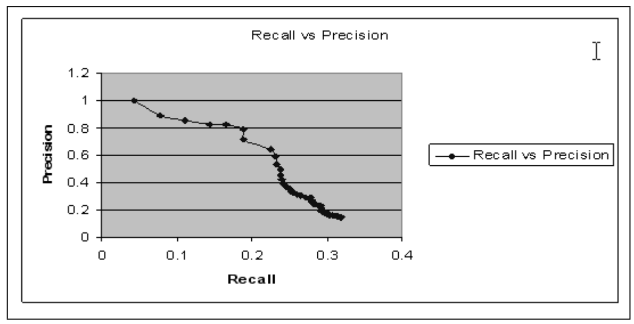


Figure 6. Recall vs. precision (Owls with 23 queries) with color proposed method.

4.4 Text-visual content-based retrieval

Our experimental result shows that our hybrid system is effective on searching image databases, especially precision was obtained by using the combination of content-based visual queries and keywords-based queries. Satisfactory retrieval results, such as retrieving all of the most relevant.

5 Conclusion

In this paper we proposed an image database searcher and indexer system that calls it ImagePickup. The aim of ImagePickup is overcome the limitation of traditional search engines and, more specifically, to find an alternative to the well known query by example paradigm. The interaction model of ImagePickup is based on the relation between images content and text can be encoded and explored. This system have three interrelated interfaces, one displaying the similarity between images by visual content, and the other displaying the similarity between words and third integrates visual similarity between images and text similarity deriv-

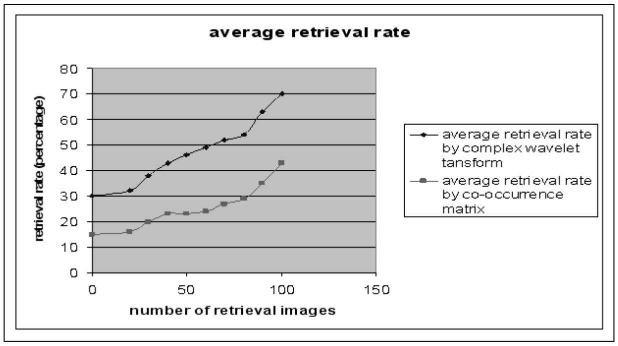


Figure 7. Retrieval rate vs. number of retrieved images (6 queries of Ducks and Owls) with texture proposed method and co-occurrence matrix.

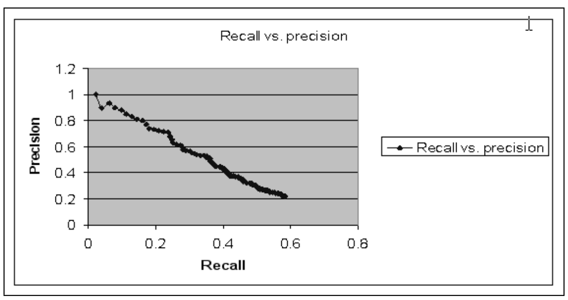


Figure 8. Retrieval rate vs. precision images (10 queries of Ducks) with texture proposed method

ing from commonality in index terms in such a way that image similarity influences text similarity and text similarity influences image similarity. Results of experiments on natural images show that higher retrieval precision and better support of user subjectivity can be achieved due to visual and text integrating.

References

[1] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom et al: *Query by Image and Video Content: The QBIC System* , Volume 28, Issue 9 (1995)

[2] J.R. Smith and C.S. Li: *Image Classification and Querying Using Composite Region Templates Computer Vision and Image Understanding* , Volume 75, pp. 165-174, Issue 1-2 (1999)

[3] S. Mukherjea, K. Hirata, and Y. Har: *AMORE: A World Wide Web Image Retrieval* , Proc. World Wide Web, vol. 2, no. 3, pp. 115-132 (1999)

[4] A. Pentland, R.W. Picard, and S. Sclaroff: " *Photobook: Tools for Content-Based Manipulation of Image Databases*, Proc. SPIE, vol. 2185, pp. 34-47, Feb. (1994).

[5] C. Carson, M. Thomas, S. Belongie, J.M. Hellerstein, and J. Malik: *Blobworld: A System for Region-Based Image Indexing and Retrieval*, Proc. Visual Information Systems, pp. 509-516 , June (1999).

[6] J.R. Smith and S.-F. Chang: *VisualSEEK: A Fully Automated Content-Based Image Query System* , Proc. ACM Multimedia, pp. 87-98, Nov, (1996).

[7] J.Z. Wang, G. Wiederhold, O. Firschein, and X.W. Sha:*Content-Based Image Indexing and Searching Using Daubechies Wavelets*," Int'l J. Digital Libraries, vol. 1, no. 4, pp. 311-328, (1998).

[8] G. Pass, R. Zabih, et al: *Comparison Images Using Color Coherence Vectors* , Proc Fourth ACM Multimedia Conference, (1996).

[9] J. Huang, S. Kumar, M. Mitra, W. Zhu, and R. Zabih:*Image indexing using color Correlogram* . In CVPR, pages 762-768, (1999).

[10] P. Rivaz, N. Kingsbury:*Complex Wavelet Features for Fast Texture Image Retrieval*, Proc. IEEE Conf ICIP, 109-113, (1999).

[11] J. Romberg, H. Choi, R. G. Baraniuk, N. G. Kingsbury : *Multiscale Classification Using Complex Wavelets and Hidden Markov Tree Models*, ICIP, (2000).

[12] Simone Santini: *The integration of textual and visual search in image databases*," In First International Workshop on Intelligent Multimedia Computing and Networking, (2000).

Using Agent Technology to Improve the Quality of Artificial Intelligence Instruction

Victor R.L. Shen
Department of Computer Science and Information Engineering
National Formosa University
Huwei 632, Yunlin, Taiwan, R.O.C.
FAX: 886-2-2205-7260; Email: victor.rlshen@msa.hinet.net *or* rlshen@nfu.edu.tw

Abstract

The author describes ways of improving the quality of AI (Artificial Intelligence) instruction, through the use of agent technology. Library agent, test center agent, ALICE (Artificial Linguistic Internet Computer Entity) agent, and student agent can be integrated into the AI-based learning environment. These agents can guide and assist students as needed, probe their understanding, and promote learning and retention. If properly designed, they can enhance student's motivation, as well. They also exploit the natural human tendency to socially respond to computing systems.

Keywords：Agent Technology, Artificial Intelligence, Intelligent Computer-Aided Instruction

1. Introduction

Currently, there has been a fast growth of interest in the Internet and the World Wide Web as vehicles for instructional delivery. Although web-based course management tools including Blackboard (http://www.blackboard.com) and WebCT (http://www.webct.com) have appeared and are widely used, the on-line courses are often patterned after classroom instruction and thus suffer from the limitations of large group instruction [2]. Actually, suppose that students are only learning their own Web-based teaching materials without any interaction with other students. The learning experience must be definitely worse than that of classroom instruction. This should produce a negative impact on student retention because students in on-line courses are poorly motivated and often fall behind.

This courseware system is intended to create a learning experience that has a significant improvement on the traditional classroom learning experience and gets closer to the ideal state of individualized instruction, since the advantages of the Web can assist students in learning anywhere and anytime. In other words, the instruction should be student-centered, and thus cause the student's motivation and foster active cognitive processing during learning. Therefore, it can provide students with individualized feedback in order to completely accomplish the learning objectives.

The rest of this paper consists of the following topics. Section 2 discusses the courseware architecture. The comparison between a controlled group and an experimental group is given in Section 3. Section 4 describes the results and discussion. And the conclusion is given in Section 5.

2. Courseware Architecture

This courseware system contains the following topics [3, 4, 6, 7, 9]:

(1) AI Introduction
(2) Search and Problem Solving
(3) Reasoning and Production System
(4) Expert Systems and Knowledge Engineering
(5) Uncertainty, Probabilistic Reasoning and Fuzzy Logic
(6) AI Programming Languages
(7) Knowledge Representation
(8) Natural Language Understanding
(9) Cognition
(10) Machine Learning
(11) Robotics
(12) Speech Recognition
(13) Neural Networks
(14) Interface/ Information Agents
(15) Multi-agent

2.1 Building the System

First of all, JRE (Java Runtime Environment) and JADE (Java Agent DEvelopment framework) (please refer to JADE installation at http://jade.cselt.it/) are needed to set up. Then we need to load down the software package AI.zip by using the Internet and decompress it into any file folder. From this step, we can obtain folder Teacher and folder Student. Folder Teacher contains all those files required for the teacher and folder Student contains all those files required for the student.

2.2 Operating the System

In this agent-based platform, teacher site is the first one to be activated. Then student site is activated anytime and anywhere by visiting the Internet system.

2.2.1 Teacher Site

Switching to folder Teacher, under the DOS mode we can type in 「java jade.Boot –gui Library:LibraryAgent TestCenter:TestCenterAgent Alice:AliceAgent」 in order to activate JADE and three agents, or just execute run.bat. After the above processes are activated, three windows will appear. The first one is JADE management window. The second one is library agent window as shown in Fig. 2.1. The third one is test center agent window as shown in Fig. 2.2. However, ALICE agent (http://www.alicebot.org/) has no window and can show some activated message under DOS mode.

Fig. 2.1 Library Agent

Fig. 2.2 Test Center Agent

Library agent is responsible for managing the textbooks used by the teacher. When the teacher wants to edit the teaching materials from the textbook, he/she should go through the following three steps: 1) Edit the contents of the textbook in HTML and upload them on a homepage. 2) Edit a setup file and upload it. 3) Add one more setup file so as to connect it to the textbook. Basically, all those HTML files can become part of a textbook. In other words, all the teaching materials must be uploaded on the web and browsed by any browser.

TestCenterAgent aims to handle the test tasks. When the teacher prepares a test, he/she must perform the following procedures. First, he needs to edit a copy of test sheet and upload it onto the homepage. Second, he needs to setup the address for the test sheet in the test center.

ALICE agent aims to handle Q&A tasks. This agent contains no management interface. All the teacher wants to manage and maintain are AI.aiml files. Basically, only executing Program B can present the answer to the question. ALICE agent is only an interface which allows Program B to get a question from other agents on the platform and passes an answer onto the asking agent.

2.2.2 Student Site

Switching to folder Student, under the DOS mode we can type in「java jade.Boot –host [Teacher's IP] –container [Student]:StudentAgent」, where [Teacher's IP] denotes the IP or domain name for teacher's computer and [Student] denotes student's name. If all the agent calling procedures are well performed and LibraryAgent is activated, a library window will appear and list all the textbooks. Once the student selects a textbook, a textbook window can appear as shown in Fig. 2.3. In addition, ALICE dialog window also appear as shown in Fig. 2.4.

Fig. 2.3 Textbook Window

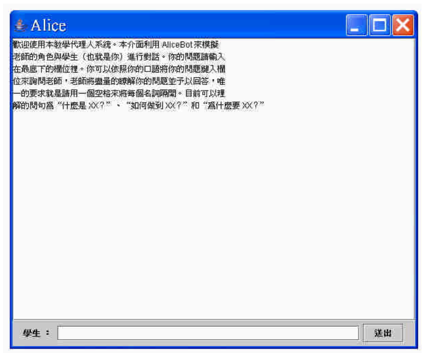


Fig. 2.4 ALICE Dialog

3. Comparison between a Controlled Group and an Experimental Group

In order to compare the outcome performance of traditional instruction with that of the proposed agent-based instruction, I select one junior class of 58 students and divide it into two groups, a controlled group and an experimental group [1, 5, 8]. Each group contains 29 students.

The controlled group of students is taught by traditional instruction and the experimental group of students is taught by the proposed agent-based instruction. Starting from the second week of this semester, I am intensively teaching Artificial Intelligence for three contiguous days using the two different instructional methods. The experimental results are summarized as follows:

For the controlled group, (1) Group Average Scores = 66.23; (2) The students feel that the teaching materials are tedious and boring; (3) Learning speed is slow and retention is ambiguous.

For the experimental group, (1) Group Average Scores = 76.25; (2) The students feel that the teaching materials are interesting and attractive; (3) Learning speed is fast and retention is obvious.

4. Results and Discussion

This courseware system presenting the following merits and results:

(1) Provides the Department of Computer Science and Information Engineering, National Formosa University, Taiwan, with a demonstrating software system on Artificial Intelligence.
(2) Provides a teacher with a computer platform to setup, save, analyze, deduce, and organize the teaching materials and files.
(3) Assists students in visualizing the behavioral performance of those complex AI algorithms.
(4) Enhances students' learning interests and efficiency.
(5)Accumulates teachers' annual teaching experiences in order to improve the quality of AI-based instruction.

5. Conclusion

This courseware system aims to assist students in clearly understanding the theory and applications of Artificial Intelligence. It obviously performs better and becomes a successful undertaking. Furthermore, by using the system we still need to cultivate students' problem solving capabilities and creativity. Finally, the student must be assigned some AI-related projects and do more exercises so as to design some creative algorithms.

Acknowledgments

The author is very grateful to Mr. Y. M. Chen, who is with the Department of Mechanical Engineering, National Central University; and to Mr. J. T. Kao, who is with the Department of Electronics Engineering, Fu-Jen Catholic University, for the tremendous amount of work that they have contributed. Also, this work was supported by the National Science Council of Taiwan, R.O.C., under grant NSC 93- 2516- S-150-002.

References

1. G. P. Baxter and R. J. Shavelson, "Science performance assessments: benchmarks and surrogates," *International Journal of Educational Research*, vol.21, no.3, pp.279-316, 1994.
2. W. L. Johnson, "Using agent technology to improve the quality of web-based education," *Proceedings of the International Conference on Artificial Intelligence in Education,* pp. 1-26, Jun. 2002.
3. R. Li, *A Theory of Conceptual Intelligence: thinking, learn, creativity, and giftedness*, Westport, Conn : praeger, 1996.
4. G.F. Luger and W.A. Stubblefield, *Artificial Intelligence*, Addison-Wesley, 3nd Ed., 2000.
5. V.M. Roth and A. Roychoudhury, "The development of science process skills in authentic contexts, " *Journal of Research in Science Teaching*, vol. 30, no. 2, pp.127-156, 1993.
6. S. Russell and P. Norvig, *Artificial Intelligence- A Modern Approach*, Prentice-Hall, 2003.
7. K.W. Tracy and P. Bouthoorn, *Object-Oriented Artificial Intelligence Using C++*, Computer Science Press, 1997.
8. J.D. Wamble, "Curriculum model for electronics technology," *Industrial Education*, vol. 9, pp. 25-27, 1988.
9. D.A. Waterman, *A Guide to Expert Systems*, Addison-Wesley Publishing Company, 1986.

Mobility in File Sharing

Hojjat Sheikhattar[1], Abolfazl Haghighat[1], Neda Noroozi[2]

1 Department of Computer Science
Shahid Beheshty University
Tehran, Iran

2 Department of Computer Engineering
Sharif University of Technology
Tehran Iran

Abstract

New advances in technology of mobile computers along with advent of wireless and mobile networking bring new capabilities, applications and concerns to the computer world. In this paper, we discussed on file sharing requirements in a mobile environment. Then, we proposed an architecture that supports mobility of hardware, data and software over the environment. Finally a sample implementation of the suggested architecture is presented.

Keywords: *File Sharing Architecture, Ubiquitous Computing, Mobile Computing, Mobile File Sharing*

1. Introduction

New advances in technology of mobile computers along with advent of wireless and mobile networking bring new capabilities, applications and concerns to the computer world. It has increased the number of mobile elements with computing power over the planet. Mobile computing [2,4,5,7,18] is the paradigm focusing on these issues.

The main aspect of mobile computing is the need of users to compute ubiquitously, anywhere and anytime [6, 21]. Transferring and sharing electronic files in a mobile environment is an important requirement of ubiquitous computing. The aim of this paper is to design a general architecture for file sharing systems that operates on mobile environments. Mobility may be logical or physical [15]. So the design of mobile file sharing system is affected by considering mobility of software agents [12] and their shared data (logical mobility) as well as mobility of hosts (physical mobility).

However due to the nature of mobility, mobile computers such a laptops, third generation mobile phones, PDAs and the like have not the similar capabilities to static hosts [9]. Wireless networks are also more error prone since air interface is used as their transmission medium. So, mobile computing has special constraints and limitations.

[18,19] characterized mobile computing by four constraints:

- *"Mobile elements are resource-poor relative to static elements."*
- *"Mobility is inherently hazardous"*
- *"Mobile connectivity is highly variable in performance and reliability."*
- *"Mobile elements rely on a finite energy source."*

Consequently, our architecture is built around a set of principles considering the constraints of mobility. Afterward, a number of design tactics and abstractions are used to describe the architecture. Mobile hospital information system (mobile HIS) is used in this paper as an example of a mobile environment with both forms of physical and logical mobility. In a hospital, patients, doctors and nurses change their location frequently, so mobile hosts may be used to help them to connect to the system ubiquitously. Besides, mobile software agents exist in the system and migrate between the hosts to gather information, help in decision making and etc.

Section 2 explains the four basic principles of the paper, section 3 describes its design issues and section 4 illustrates our sample implementation of the architecture.

2. Principles

Our architecture is based on a set of principles to satisfy the additional requirements of the mobile file sharing system. These principles are core of our design phase and have designated as a supplement to notion of typical file sharing systems.

1. Support for heterogeneous environments with adaptable behavior

Since nomadic user of the future needs pervasive information access, mobile platforms are supposed to connect to the network ubiquitously. So, the mobile file sharing systems should support several access technologies including wireless LAN, short range radio and packet radio and ought to work in heterogeneous environments. It means, in contrast to fixed networks, mobile file sharing systems should operate on multiple networks. For example, in a mobile HIS that two hosts may have different connections through the global wireless network (i.e. UMTS) or the hospital wireless network or a transient ad-hoc network, the file sharing system should allow authorized users to select their desired network to transmit the shared files.

On the other hand, as a result of hand-over of mobile users between diverse networks as well as alteration in connectivity of the networks - and in general due to dynamic change of the computational context - designers have the tendency to use adaptive[1,8,17,19] systems for mobile environments. Adaptability is based on the rule that when the situation changes, then the behavior of the system adjusts in line with predefined or instant policies of the user. Network-adaptation, application specific adaptation and adaptation in content transmission are some examples of adaptation strategies have been researched [14]. For file sharing system, we decided on application aware adaptation [11,16] which can operate on the above strategies. Assume that a nomadic user move to a low bandwidth and costly network when he is downloading a large file, in this case the system should not continue file transfer stupidly. It should make a decision through predefined policies or ask the user to decide. So our architecture should support heterogeneity in file transmission plus methods for adaptability of the system with the environment changes.

2. **Decrease the number of unnecessary and inefficient file transfers**

In fixed networks it is ordinary to download a file from a host incorrectly just because of name similarity. These events are not desirable in mobile environments while the cost of file transmission in mobile networks is much more than fixed ones. In addition to unnecessary file transmissions, inefficient file transfers may appear. Inefficient file transfers may happen due to lack of information about the shared files and the sharers. As an example, the file may be downloaded from a host with poor connectivity while the file has been shared by its original source with better connectivity. Another example is starting file transmission from a host which is going to leave the network in minutes.

Both incorrect and inefficient file transmissions are occurred owing to lack of information. Indeed, in fixed networks as the speed is high and the cost of communication is low, metadata about shared files are limited. However, in mobile environments due to dynamic behavior of the mobile nodes as well as high cost and low speed of the network, sufficient metadata attached to shared files required to decrease the number of incorrect and inefficient file transmissions.

3. **Avoid unsolicited waits**

In wireless networks, absence of a host is common. So, when a file is requested from an absent host, the requester waits for its presence. Usually in typical file sharing systems, the requester will not be informed at the time the sharer come back to the network. So, the requester uses the polling method to check the presence of the sharer. Sometimes, this leads to a long delay before file transmission. Since, the requester itself may not be within reach when the sharer returns to the network. Another obstacle is that sharers will not be informed about file requests sent to them in their absence time. This cause unnecessary waits too.

To avoid mentioned unwanted waits, the file sharing system should support a scheme to inform waited hosts when the wanted host returned to the network. It also should maintain asynchronous file requests.

4. **Use simple and consistent naming system**

In a mobile environment, with both logical and physical mobility, everything is dynamic. Mobile hosts are moving between networks and mobile agents are moving between hosts. Users are also mobile and may even change their host to have ubiquitous information access. Naming elements in such an environment is an interesting research topic. For file sharing system, we need a consistent naming system, in which shared files have persistent access names with low degree of complexity. In our architecture we employ a simple but effective idea to fulfill the requirements of mobile file sharing system.

3. Design Issues

In design of the mobile file sharing system, we used a number of tactics [3] to satisfy the requirements and quality attributes of the system. These tactics are a collection of related works that are managed with a predefined policy based on the principles mentioned in section 2.

3.1 Tactics

1. **Layered Design**

Like most of distributed systems, our architecture exploits a layered system. It has three major layers and each layer is almost separate from the others. The lowest layer is *relay layer* which is responsible for data communication; the median layer is *coordination layer* which provides the abstraction of a shared space and helps the peers to convey file sharing control records; the highest layer is *file sharing managerial layer* which determines the arrangement and meaning of file sharing control records. This structure supports heterogeneity, as the relay layer communication method could change without modification in upper layers.

2. XML structures as descriptive metadata

As described in second principle of section 2, maintaining descriptive metadata will reduce the number of inefficient and incorrect file transfers. We used two plain xml structures. One for publishing a shared file metadata (FXMD) and the other for describing status of a sharer host (HXMD). The former will be extended by the latter to form the shared file effective metadata (EXMD). Just as a sharer receives a file request, it generates effective metadata for the file and passes it to the requester. Figure 1 represents an instance of a shared file effective metadata. The brief description of the file, the original source (or a fixed host) of the file, the exact size of it, the possibility of segmentation of the file are some information that may be placed in FXMD. Estimated time the host stays in the network, estimated time the mobile agent remains on the host, the supported file transfer methods, the quality of shared connections are some information passes with the HXMD.

```
<EXMD>
   <FXMD>
       <filename>P028CH </filename>
       <source> HIS File Server 2
       </source>
       <description>
       History of cardiograph outputs for
       patient 028 (Sara Hosseini). From
       2004/08/10 to 2004/11/10
       </description>
       <size> 15 KB </size>
       <dividable> no </dividable>
   </FXMD>
   <HXMD>
       <connections>
         <C><N> TCI </N><Q> average </Q>
         <C><N> LHIS </N><Q>excellent </Q>
         <C><N> A12 </N><Q> not good </Q>
       </connections>
       <EDD> 14:20 </EDD>
       <EAD> N </EAD>
   </HXMD>
</EXMD>
```

Figure 1. Sample Effective Metadata

3. Application aware adaptation

Adaptation stands for dynamic balance between two paradigms, autonomy of mobile hosts and interdependence of them. The first has the advantage of being free from network connectivity concerns but suffers form resource poverty of mobile stations. However the interdependence paradigm is vice-versa. Satyanarayanan delimits the range of strategies for adaptation by two extremes, as shown in Figure 2. "At one extreme, adaptation is entirely the responsibility of individual applications. … The other extreme of *application-transparent* places entire responsibility for adaptation on the system. … Between these two extremes lies a spectrum of possibilities that we collectively refer to as *application-aware adaptation."*[17]

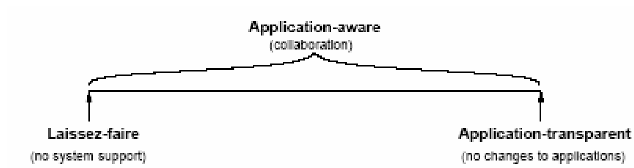


Figure 2. Range of adaptation strategies [20]

In our architecture, application aware adaptation is exploited to let the file sharing system and its users tune the behavior of mobile hosts.

4. Utilize listeners, status info and ASYN requests

An effective approach to avoid unsolicited waits is taking advantage of asynchronous communication. In our design, this approach is employed in three ways.

Putting status info of the peers in the shared space

Before attempt to file transfer, the requester checks for the presence of the sharer host. Putting status info of the hosts in the shared space could reduce the cost of communication as well as waiting times. Status info of a host is updated when it decides to leave the network. Moreover, on revisiting the network it updates its status info in the shared space and changes the status to present.

Setting listeners for the peers when they are absent

When a sharer host is absent, requesters can set listeners for it in the shared space and ask the system to notify him when the sharer host revisits the network. Using this technique the requester informs about sharer return immediately and makes it possible to apply for the shared file instantly.

Using ASYN file requests if a file sharer is absent

When a sharer host is absent the requester can put an ASYN request for a file in the shared space and ask the sharer host to send him the file. The ASYN request may suggest asynchronous or synchronous file transfer methods for sending the file. That is, if the requester predicts an unstable state, it asks for asynchronous transmission and in contrast, stable state prediction may cause a synchronous file transmission.

In mentioned techniques, the concept of shared space is an essential part of the communications. This concept is realized by the coordination layer as stated before. The coordination layer divides the shared space into two parts, a *public area* which is common between all hosts and a *private area* which holds a message inbox for each host. Listeners and ASYN file requests are put in the private area and status info records are located at public area.

5. Two-level naming system

In a mobile file sharing system hosts migrate between networks and agents and their associated files migrate between hosts. Users also change their hosts and their related data according to required computing context. In such environments with high degree of mobility, the traditional naming systems are not well suited [20]. So addressing the files and locating them with a non-complex structure that runs on mobile hosts efficiently is a challenge. Our solution uses a two level naming system. In which, *access names* as permanent and human-friendly names are mapped to *address names* as temporal and location dependent names. The idea is simple. Identifiers of the sharers are the sole names that are not changing in this environment. File sharers are either mobile users or mobile software agents. Both groups have unique and human-friendly identifiers. So access names consist of sharer identifiers as the first segment and file names as the second segment. For example in a mobile hospital information system if the user "Nurse103" is going to share the file "BloodPressureA29" the shared file access name will be "/Nurse103/BloodPressureA29". It is emulates a regular file sharing system wherein each user has a separate directory and there are some automatic mount points to user directories. This scheme leads permanent names that do not change by migration of software agents and users. On the contrary, address names consist of three segments, network name, host network address and file address. Network name indicates the last network visited by the mobile host, host network address is the address of the host in the last network addressing system and the file address is the address of the file in the host addressing system. This scheme permits migration of mobile hosts between networks and user data between hosts without any change to access names. The advantage of this naming system is acquiring name transparency with a direct single-level mapping instead of using complex multi-level mappings which do not fit into the constraints of mobility.

3.2 Structure

The first level decomposition [3] of mobile file sharing system based on the mentioned principles and tactics is shown in figure 3.

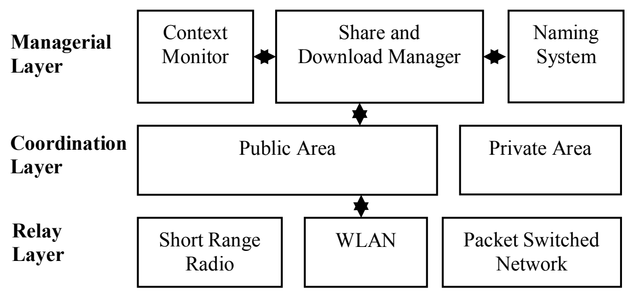


Figure 3. First Level Decomposition of the System

Below is brief description of the components of the managerial layers.

Context Monitor:

Context monitor is responsible for changes of computing context including quality of network connection, remaining battery, hosts physical location and etc. Monitoring the computing context in managerial layer makes dynamic adaptation of the application possible.

Share and Download Manager:

Share and download manager is in charge of the main procedures of the file sharing system including procedures of sharing and downloading files, handling XML metadata structures, listeners, status info records, and ASYN requests.

Naming System:

Naming System manages the two-level naming scheme. It uses the public area of coordination layer to store the names. When a user changes his host, or a software agent migrate from a host to another, the naming system of the new host updates the address of the shared files in the public area. When a mobile host enters to a new network or changes its network address the update is done by naming system of the mobile host itself.

4. Implementation

The architecture presented in this paper is general and several implementations can realize it. For example the concept of the shared space and coordination layer fit to different patterns in line with the requirements of the system. That is this architecture works on a variety of environments. For instance, at one extreme if a fixed server is available in the system and supports mobile hosts with a good connection (like hospital information system) then the shared space could be deployed on the fixed server. The other extreme is ad-hoc networking, whereas there is no fixed infrastructure and the networks are short-lived. In ad-hoc networks, coordination layer uses the concept of virtual shared space and emulates the shared space features for the upper layer.

To verify the correctness of presented architecture, we implemented a prototype mobile file sharing system in java. It conforms to our design issues and principles and well suited to the mobile environments. The coordination layer in sample implementation is based on LIME[10,13], a Linda-like coordination system with mobility support. The context monitor of the implementation is straightforward and just captures the predefined values to inform the managerial layer about the context. The naming system is as described in the paper, but it also uses hashing to improve performance. Share and download manager has a simple user interface and an engine to handle the sharing and downloading procedures. The sample implementation shows that the notion of the proposed architecture is correct and it is feasible to realize it in mobile environments.

5. Conclusion and Future Work

It can be seen from this paper that using new models for file sharing in mobile environments, can improve usability, performance and effectiveness of file sharing systems with mobility support. In our recent work on mobile file sharing systems, we employed an architectural style and several design techniques to fulfill the requirements and quality attributes of the mobile file sharing system. Adaptation as a key solution for mobility problems, asynchronous communication and supplementary information as two ways for reducing unnecessary waits, and innovative naming system to simplify addressing the mobile units are some design techniques exploited in our model.

In our ongoing research we plan to investigate further on adaptation and design more clever applications. To address this issue, we have divided our domain into two areas, the first denotes to pure mobile networks in which there are not auxiliary fixed hosts and everything is mobile, and the second denotes to common mobile networks that benefits from supporting fixed hosts. The next studies can focus on specific models and implementations for each area as well as working on flexible systems to cover both areas with the ability of switching between them.

References

[1] A. Al-bar, I. Wakeman, *A Survey of Adaptive Applications in Mobile Computing.* Proc. 21st International Conference on Distributed Computing Systems Workshops, pp. 0246, Apr. 2001.

[2] D. Barbara. *Mobile computing and databases - a survey*. IEEE Transaction on Knowledge Data Engineering, pp. 108-117, Jan. 1999.

[3] L. Bass, P. Clements, R. Kazman. *Software Architecture in Practice.* Second Edition, Addison Wisely, 2003, cited on pages 100, 160.

[4] Y. Chen, C. Petrie. *Ubiquitous Mobile Computing.* IEEE Internet Computing, pp. 16-17, Mar/Apr. 2003.

[5] Chlamtac, J. Redi. *Mobile Computing: Challenges and Potential.* Encyclopedia of Computer Science, 4th Edition, International Thomson Publishing, 1998.

[6] T. T. Drashansky, S. Weerawarana, A. Joshi, R. A. Weerasinghe, E. N. Houstis. *Software architecture of ubiquitous scientific computing environments for mobile platforms*. ACM Journal on Mobile Networks and Applications, 1(4), 1996.

[7] T Imielinski and B. R. Badrinath. *Mobile wireless computing: Challenges in data management*. Communications of the ACM 37(10), pp. 18-28, Oct. 1994.

[8] R. Katz. *Adaptation and mobility in wireless information systems*, IEEE Personal Communications, pp. 6-17, 1994.

[9] C. Mascolo, L. Capra, S. gachariadis, W. Emmerich. *XMIDDLE: A DataSharing Middleware for Mobile Computing*. Wireless Personal Communications, pp. 77-103, 2002.

[10] A. Murphy, G Picco, and G. Romjan. *Lime: A coordination middleware supporting mobility of hosts and agents*. Technical Report WUCSE-03-21, Washington University, Department of Computer Science, St. Louis, MO (USA), 2003.

[11] B. D. Noble, M. Price, and M. Satyanarayanan. *A programming interface for application-aware adaptation in mobile computing*. Computing Systems, 8(4), 1995. Also as technical report

CMU-CS-95-119, School of Computer Science, Carnegie Mellon University.

[12] V. Pham, A. Karmouch. *Mobile Software Agent: An Overview.* IEEE Communication Magazine, p.26-37, July 1998.

[13] G. Picco, A. Murphy, and G. Roman, *LIME: Linda meets mobility*, in International Conference on Software Engineering, pp. 368--377, 1999.

[14] C. Policroniades, R. Chakravorty, P. Vidales. *A Data Repository for Fine-Grained Adaptation in Heterogeneous Environments.* Proc. Third ACM international workshop on Data engineering for wireless and mobile access, ACM Press, New York, pp. 51-55, Sept. 2003.

[15] I. Satoh, *Physical Mobility and Logical Mobility in Ubiquitous Computing Environments*, Proc. Conference on Mobile Agents (MA'02), LNCS, Vol. 2535, pp.186-202, Springer, 2002.

[16] M. Satyanarayanan, B. Noble, P. Kumar, M. Price. *Application-Aware Adaptation for Mobile Computing*. School of Computer Science, Carnegie Mellon University.

[17] M. Satyanarayanan, *Fundamental Challenges in Mobile Computing.* Proc. Fifteenth annual ACM symposium on Principles of distributed computing, ACM Press, New York, pp. 1-7, 1996.

[18] M. Satyanarayanan. *Mobile Computing.* URL: www.computer.org/computer, pp. 81-82, 1993.

[19] M. Satyanarayanan*. Mobile Information Access.* IEEE Personal Communications, February 1996.

[20] A. Tanenbaum, M. Van Steen*. Distributed Systems, Principles and Paradigms.* Prentice Hall, 2002, cited on page 210.

[21] M. Weiser. *The Computer for the Twenty-First Century*. Scientific American, pp. 94-104, Sept. 1991.

A BINARY SOAP-BASED MESSAGING INFRASTRUCTURE FOR MOBILE MULTIMEDIA SESSIONS

Bjorn Muylaert	Stijn Decneut	Matthias De Geyter
VRT	IMEC	VRT
Reyerslaan 52	Kapeldreef 75	Reyerslaan 52
1043 Brussels, Belgium	3001 Leuven, Belgium	1043 Brussels, Belgium
bjorn.muylaert@vrt.be	stijn.decneut@imec.be	matthias.degeyter@vrt.be

Abstract

Multimedia streaming to mobile devices is gaining popularity. In order to provide a high Quality-of-Experience to users of such devices, adaptation and caching mechanisms need to be provided at various places in the network. Such mechanisms can only perform well when they have sufficient metadata on the media, network, user and device at their disposal. In this paper, we identify the requirements that are posed on metadata exchange mechanisms in a mobile context. Since none of the available mechanisms meet the requirements on a satisfactory level, we propose a new binary XML-based messaging infrastructure that does perform as required. Our experiments show that this infrastructure offers the needed functionality, while consuming less bandwidth and power than current mechanisms.

1. Introduction

Mobile devices are faced with continuously changing circumstances, such as available bandwidth and environmental conditions. Furthermore they have to deal with limited battery and calculation power. When such devices are used to produce and/or consume multimedia content, changing circumstances impact the perceived quality of the multimedia significantly. To deal with these variations, adaptation and caching strategies may be provided at several places in the network. In order to find optimal solutions, such strategies need to gather information from different parties (user, device, network) at various points in the network [6] [8].

We have developed a binary XML-based messaging infrastructure that combines a number of recent standards to meet the requirements of metadata exchange with mobile multimedia devices. The infrastructure comprises a formal definition and binary encoding of the XML messages, an extensive API and two reference implementations.

In this article, we elaborate on the specific requirements for mobile multimedia terminals, detail the technologies we integrate in our messaging infrastructure, describe our experiments and evaluate the results.

2. Requirements on exchange of metadata

Various XML-based languages that describe metadata are gaining momentum. Popular examples are TV-Anytime metadata [2], CC/PP [10] and numerous MPEG-21 tools [3]. Figure 1 shows the types of metadata that we take into consideration. When mobile terminals need to exchange such metadata with other parties in a network, a widely accepted, flexible yet efficient and effective communication mechanism is required.

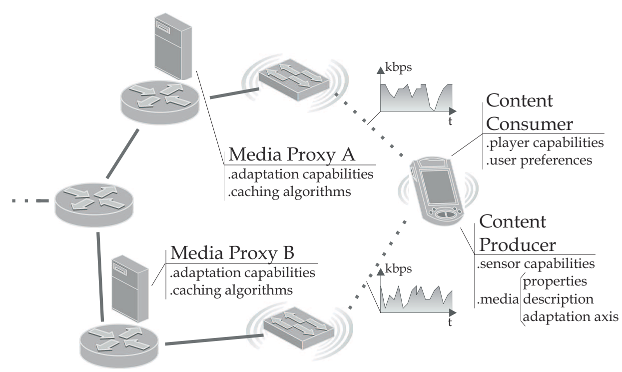


Figure 1. When a multimedia terminal travels through wireless networks, various time-dependent metadata from media proxies, network conditions and the terminal needs to be communicated effectively.

When focussing on low-power, mobile devices, any communication mechanism needs to be able to transport the metadata efficiently. A high level of efficiency is obtained by sending only relevant and new information, while minimizing the protocol overhead and the amount of useless metadata transport, by preventing both the retransmission of already known data and any data transmission which the receiver cannot correctly perceive. This also means that the mechanism should be able to send fragments of metadata when only a partial update of information is required. The efficiency of the communication mechanism furthermore lies in its ability to be employed by mobile devices with minimal power consumption.

An effective communication mechanism that operates in an environment as shown in Figure 1, needs to bring across information in varying and complex circumstances with a high degree of reliability. The mechanism has to operate in a continuously changing combination of environments that differ in security, (service and network) reliability and availability. This means that it has to provide its own control over communication-related aspects such as encryption, redundancy and redirection. Effective communication is also facilitated by the availability of formal mechanisms to describe available services and transported messages, since this creates a common understanding of terminology by the parties involved.

The flexibility of a communication mechanism lies in the way it is able to cope with changes in types of metadata it is able to convey, network environments over which it can be transported and capabilities of the devices by which it is employed. Underlying network environments can range from error-prone wireless to wired broadband. Target devices include anything from low-power mobile terminals to enterprise servers.

Shortcomings in efficiency and effectiveness typically occur when a communication mechanism is employed in circumstances other than anticipated by its creators. To overcome such shortcomings, flexible mechanisms may allow future extensions and backward compatible modifications. However, such extensions and modifications often do not fit nicely in the original concept of the technology, and as such become a sub-optimal solution. For instance, HTTP, which is by origin a client-server communication mechanism, was not devised with multi-node communication in mind, yet can be used for it when other mechanisms are built on top of it. One way to accomplish this is by creating a client-server connection from each node in the network to the next. That way, the application behaves as a multi-node system while in fact message forwarding is used at each node.

Currently used protocols for aspects of multimedia management, such as session initiation (e.g. SIP [7]) and delivery control (e.g. RTSP [7]) do not meet the posed requirements to a satisfactory degree. These protocols were developed prior to the emergence of mobile multimedia sessions, and were as such not developed with mobile applications in mind, and thus suffer from a combination of these drawbacks: they communicate (XML-based) metadata inefficiently; they are ineffective in mobile multimedia environments (e.g. unable to handle changing network topologies); there is no native support to overcome the previous two in a flexible way.

RTSP for example, is an effective and efficient protocol for the control of streaming multimedia delivery in a simple client-server scenario without mobility or intermediate adaptation nodes. However, when used by mobile multimedia terminals, its effectiveness is diminished due to its HTTP-like client-server interaction. This construction hampers mobility, where sessions migrate between servers, networks and even terminals. With RTSP, it is hard to add and to formally describe the functionality intermediate nodes (i.e. media proxies) offer and the messages they use to communicate. Yet, when such aspects change frequently, this is crucial to achieve effective and efficient communication. Without a formal description of both messages and services, unnecessary communication is practically unavoidable, as our experiments in Section 4.2 will illustrate.

3. Messaging infrastructure

We have developed a multimedia metadata messaging infrastructure that meets the requirements we have set in the previous section. To facilitate widespread acceptance, we have based it on recently developed standards in the field of XML-based communications. We have provided APIs that allow construction and handling/parsing of messages. Furthermore, we provide two reference implementations of the API, optimized for either low power (embedded) or large scale server systems. In order to allow a high efficiency in bandwidth and power consumption, we provide a mechanism to exchange messages in a standard binary format. We use the name *Liquid SOAP (LiqS)* for our messaging infrastructure with integrated technologies and *LiqSb* for the binary encoded messages.

3.1. Messaging protocols

We have integrated the most flexible and effective XML-based messaging protocol available: SOAP 1.2 [9], a W3C Recommendation. In combination with Web Service Addressing (WS-Addressing) [11] and SOAP TCP binding it proves to be an excellent tool for the exchange of XML data in complex environments. Figure 2 shows the envelope of a message that combines these three technologies.

Unlike the popular SOAP HTTP binding, which always results in a client-server type request-response mechanism, our SOAP TCP binding allows a range of commu-

```
<?xml version="1.0" encoding="utf-8" ?>
<soap:Envelope
xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
xmlns:wsa="http://schemas.xmlsoap.org/ws/.../addressing/">
 <soap:Header>
  <wsa:Action>urn:play</wsa:Action>
  <wsa:MessageID>uuid:9ffc2332-82b1-4c9...</wsa:MessageID>
  <wsa:ReplyTo>soap.tcp://clientplayer:8081/</wsa:ReplyTo>
  <wsa:To>soap.tcp://mediagateway/sessionservice</wsa:To>
 </soap:Header>
 <soap:Body> ... </soap:Body>
 <soap:Fault> ... </soap:Fault>
</soap:Envelope>
```

Figure 2. Example of an empty message envelope, which is a standard SOAP envelope that contains WS-Addressing information. '...' Indicates characters were removed for clarity.

nication and choreography models, such as *fire-and-forget* and *request-notify*. Using our TCP binding, a server in a network can easily send messages to a (mobile) device, as opposed to the widespread HTTP binding where this could only happen if the client device would regularly send polling messages to the server or host a HTTP server, both unacceptable power hungry mechanisms. SOAP TCP binding allows messages to be sent to SOAP nodes other than their final destination, requiring formal mechanisms for forwarding and redirection. We integrated WS-Addressing which offers exactly the needed functionalities: message identification, message reference, sender and receiver identification, and more.

3.2. Binary XML

XML-based messaging frameworks such as ours offer significant advantages with regard to many aspects, such as flexibility, clarity, consistency and rigidity. However, their default text-based transport places a heavy burden on the available bandwidth and processing power of communicating devices. During the development of the MPEG-7 standard, a binary format [5] was devised, that meets the specific requirements of multimedia terminals while preserving the many advantages of XML-based communication. Apart from a performing schema-based data compression and power efficient parsing on a bit-stream level, the format allows fast random access, fragmenting, streaming and selective updating of XML documents, making it the first format that satisfies all of the requirements identified by MPEG [4].

Since our SOAP messages are entirely standard XML based, we were able to encode them using the MPEG-7 binary XML mechanism. We use the Expway BinXML™software, a standard compliant implementation [1].

3.3. API

We have developed APIs to send and receive our messages. Since we target a range of devices from low-power terminals to high-end servers, we have built two reference implementations.

One implementation is optimized for low-power and high efficiency but it provides for one simultaneous incoming/outgoing connection only. It is built without assuming functionality or libraries that are specific to a development environment. We perform all message handling, manipulation and composition directly in our software.

The other reference implementation we provide, is built using well-accepted server technology from Apache and Sun Microsystems. It is able to handle large amounts of connections, while focusing less on power consumption.

4. Evaluation

To compare the behaviour and performance of our messaging infrastructure with another mechanism, we need to build two identical working environments such as the one in Figure 1, one using our mechanism and one using the other. We have chosen to use RTSP as a case study. RTSP is a popular and well optimized protocol that performs very well in traditional client-server environments. As an example, Figure 3 shows a typical RTSP Play message, which a client sends to request a server to start streaming the media of a prepared session. RTSP defines nine more messages and a large set of response/error codes, similar to those of HTTP.

```
PLAY rtsp://...sample_300kbit.mov/track... RTSP/1.0\r\n
CSeq: 4\r\n
Range: npt=0.000000-66.470000\r\n
x-prebuffer: maxtime=2.000000\r\n
x-transport-options:late-tolerance=10\r\n
Session:3636553809122718327\r\n
User-Agent:QTS(qtver=6.4;os=WindowsNT 5.0Ser...)\r\n
\r\n
```

Figure 3. Example of a RTSP Play message. '...' Indicates characters were removed for clarity.

We have build XML-based counterparts for all RTSP messages, described by formal WSDL descriptions. The messages and data types are described in an XML Schema, named *LiqSRTS*, which stands for *Liquid SOAP, Real Time Streaming*. For the purpose of this evaluation, we placed exactly the same information in our messages that is conveyed in their respective RTSP counterparts, even though our messaging framework does not require such a direct mapping. Figure 4 shows a one-to-one translation of the RTSP Play message in XML.

```
<Play xmlns:liqs="http://www.xmt.be/.../LiqSRTS/">
 <user>
  <name>QTS</name>
  <properties>
   <version>6.3</version>
   <os>Windows NT 5.0 Service Pack 4</os>
  </properties>
 </user>
 <session>
  <id>3636553809122718327</id>
  <options>...</options>
  <media>
   <uri> liqs://...sample_300kbit.mov/trackID=3</uri>
   <range type="npt">
    <start>0.000000</start>
    <stop>66.470000</stop>
   </range>
  </media>
 </session>
</Play>
```

Figure 4. Example of the body of an RTS Play message, translated from the RTSP Play message of Figure 3. '...' Indicates characters were removed for clarity.

4.1. Functional performance

Using our reference implementations to handle the messaging, we developed a small software library that translates RTSP and our messages in both directions. Using that library we were able to rapidly develop both a Liqs compliant player and server. In order to create a Liqs compliant server we wrapped an existing Darwin Streaming Server (DSS) to communicate using only our messaging framework. The same library, although using a different message direction, enabled to create a Java-based media player. With the available components we built two test setups: one entirely LiqS based (LiqS DSS, Java media player and LiqS proxies which we developed ourselves) and one entirely RTSP based (DSS, QuickTime 6.4 and proxies, compiled from RealNetworks RTSP Proxy Reference Kit 2.0). Some of the most eye catching conclusions of our experiments are detailed hereafter.

Already during the development and integration phase of our experiments, some clear advantages of our messaging framework surfaced. Because we have formal XML Schema descriptions of all LiqS messages and included data types, we are able to reuse third party message parsing and generating components, knowing from start that they handle the messages correctly, even validating them explicitly using XML Schema. When building RTSP parsers and generators, on the other hand, much development effort is needlessly spent on the error prone character set level.

The real merits of our messaging infrastructure, however, became clear once we tried to build the setup as described above. Our LiqS player was able to send messages to a proxy in the network, knowing that they would be forwarded to the correct destination, as detailed in the WS-Addressing information. With our RTSP setup, however, after rewriting the proxy software so that multiple proxies could be chained, we needed to add proprietary information to the messages which describe the required path explicitly. Such a non-standard mechanism offers no guarantee that the information will be interpreted correctly at the receiving end. Even then, we were not able to provide the same functionality with our RTSP setup, unless we stepped away even further from the standard, since RTSP has no standard accepted mechanism to indicate the location where the RTP stream should be sent. RTSP does have a standard way to tell the server where the stream is to be sent, by the "description=" key which can be included in the "Transport:" field of the SETUP message. However since it is a candidate for distributed denial of service attacks, it is recommended that implementations ignore it unless the requesting client is authenticated using HTTP authentication mechanisms, and to use the client's IP address otherwise. We are convinced that those security issues should be handled at another level, as WS-Security does for Web Services.

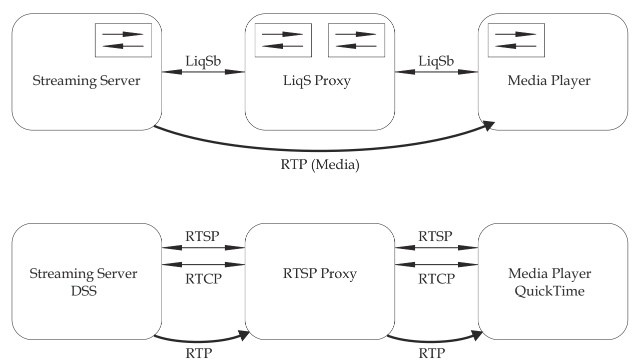


Figure 5. Comparison between simple streaming server, proxy-, player-setup using LiqS or RTSP as messaging protocol.

The lack of RTP redirection forced us to route all RTP-streaming packages through the RTSP proxies, while the Liqs-setup didn't suffer from this architectural choice, enabling direct streaming from the server to the client, as can be seen in Figure 5.

We included XML-based metadata (user preferences and MPEG-7 media information) in some messages. Our approach enables this seamlessly, simply by adding the metadata as an extra subtree in a message. Using the XML Schema, a recipient of a message is able to validate and interpret the information. RTSP provides no such straightforward mechanism for the inclusion of XML or any other alien data in messages. For every piece of XML data, we had to calculate its length and include it as foreign data in a message, where it loses its meaning.

4.2. Bandwidth consumption

We have translated a set of typical small RTSP messages in our messaging framework, in the same way as illustrated in Figure 3 and Figure 4. The messages were taken from a basic test that we performed by streaming a Quicktime demo movie from DSS with a Quicktime player.

We measured the size of both variants of the messages, as they are sent to the network stack (i.e. without overhead from lower layers). Figure 6 shows the messages with the respective sizes we measured. While RTSP was designed with conciseness in mind, our binary XML messages are often significantly smaller. The bottom row named 'Total', which combines the amount of data sent to start the streaming of a demo movie containing 2 tracks, one audio and one video, gives an overview of the typical message traffic between client and server. The schema column shows the size of the compiled binary schemas, which are needed by any recipient of the messages to be able to parse the messages. The number given in the trade-off column indicates the number of messages of that particular type would need to be exchanged per schema update before the overhead of sending the schema is compensated. This means that, when at least a 'trade-off' number of messages is sent per schema update, LiqSb requires less bandwidth than RTSP. We can safely assume that schema updates should rarely take place, since they are only required when the actual format of the messages changes.

Message	RTSP	LiqSb	schema	trade-off
Describe	211	160	1582	31
Description	817	337	2160	5
Setup	327	232	1666	18
Response	511	189	1795	5
Options †	1550	0	0	-
Play	275	209	1671	26
Ok	259	145	1599	14
Total	4788	1693	10473	4

Figure 6. Measurements of the size (in bytes) of typical RTSP messages, equivalent LiqSb messages and the binary schema.

As marked by †, in our RTSP setup the DSS sent an 'Options' message which our Quicktime player did not understand. This large message would not have been transmitted, if the client was able to provide a description of the messages it can handle through some sort of formal mechanism. With our messaging framework, such useless message transmission can be eliminated, which can result in significant bandwidth savings.

The actual media data that is transmitted from server to client requires significantly more bandwidth than these messages, as they are very basic and small. However, when large amounts of metadata need to be exchanged, which is often the case with MPEG-7 and MPEG-21, the bandwidth that is saved by our messages becomes important.

5. Summary and conclusions

We have developed a binary XML-based messaging infrastructure that combines new standards in the field of XML messaging to meet the requirements of metadata exchange with mobile multimedia devices. The infrastructure comprises a formal definition and binary encoding of XML messages, an extensive API and two reference implementations.

Our experiments show that our infrastructure fits much nicer in complex multimedia environments where mobile devices wish to exchange metadata. We measured the bandwidth implications of our approach, and even in a simple scenario, our messaging mechanism saves bandwidth over RTSP.

6. Acknowledgments

This work is a result of a collaboration between VRT, IMEC, UGent and VUB. This is one of the E-VRT projects funded by the Flemish government.

References

[1] *Expway BinXML*. http://www.expway.com/.
[2] *TV-Anytime Forum*. http://www.tv-anytime.org/.
[3] ISO/IEC JTC1/SC29/WG11 21000. *Information technology – Multimedia Framework (MPEG-21)*.
[4] ISO/IEC JTC1/SC29/WG11 N4320. *MPEG-7 requirements*, July.
[5] U. Niedermeier, J. Heuer, A. Hutter, W. Stechele, and A. Kaup. An mpeg-7 tool for compression and streaming of xml data. In *Proceedings of the 3rd IEEE International Conference on Multimedia & Expo*, pages 521–524, 2002.
[6] P. Schojer, L. Böszörmenyi, H. Hellwagner, B. Penz, and S. Podlipnig. Architecture of a quality based intelligent proxy (qbix) for mpeg-4 video. In *Proceedings of the 12th International World Wide Web Conference*, pages 394–402, 2003.
[7] H. Schulzrinne. A comprehensive multimedia control architecture for the internet. In *Proceedings of the International Workshop on Network and Operating System Support for Digital Audio and Video*, pages 65–76, May 1997.
[8] P. Soetens, M. D. Geyter, and S. Decneut. Multi-step media adaptation with semantic web services. In *Proceedings of the 3rd International Semantic Web Conference*, 2004.
[9] W3C. *SOAP Version 1.2*, 2003. W3C Recommendation.
[10] W3C. *Composite Capability/Preferences Profile*, Jan. 2004. W3C Recommendation.
[11] W3C. *Web Services Addressing*, Dec. 2004. W3C Working Draft.

HyperGuide: a context-aware semantically interoperable multimedia application for the fruition of cultural heritage

Giorgio Ventre
Universitá degli Studi di Napoli ”Federico II”
Napoli, Italy
giorgio.ventre@unina.it

Francesco Gragnani and Vincenzo Masucci and Claudio Nardi and Vincenzo Orabona
Consorzio Campano di Ricerca per l’Informatica e l’Automazione Industriale
Piazza Enrico Fermi 1, Portici, Italy
c.nardi@criai.it

Abstract

In this paper we present a semantically interoperable software application, codenamed HyperGuide, for supporting the fruition of cultural heritage through proposition of context-aware multimedia content. Semantic languages have been used to achieve semantic interoperability for data and for some subsystems. The application was designed and developed at the CRIAI research lab in Portici (Naples), Italy within a research program funded by the Italian Education, University and Research Ministry. The program is currently entering its experimental phase when the application is being adopted in one of the archeological sites of the Soprintendenza archeologica of Pompei. In HyperGuide context is a multi-dimensional entity which refers to three aspects: the user’s profile, the user’s device capabilities, and the user’s real-time geographical position. Multimedia content is described and stored through standard ontologies and languages. The paradigm of device independence is exploited to provide a standard description of the device’s capabilities while the user’s preferences are captured through a lightweight registration process. When the user accesses the system, his/her position is determined through a fine-grained indoor and outdoor location system. At this point, multimedia content is retrieved and content presentation is automatically adapted based on the context.

1. Introduction

Cultural tourism is a growing segment of the tourism industry and the one that offers the best opportunities to local communities in terms of economic development and preservation of local landscapes and lifestyles. However, the visitor to cultural heritage sites is often frustrated at the lack or inaccuracy of the relevant information. As a result the visit is not tailored to the tourist sensibility and education and this, in turn, may take him/her away from this kind of experience. In recent years though, rapid advances in technology, from telecommunications to multimedia, hardware, speech synthesis and recognition, and graphics have proven useful in documenting, preserving, and disseminating our valuable culture heritage. In this paper we present a semantically interoperable software application, codenamed HyperGuide, for supporting the fruition of cultural heritage through proposition of context-aware multimedia content. The application was designed and developed at the CRIAI research lab in Portici (Naples), Italy within a research program cosponsored by the Italian Education, University and Research Ministry [1]. The program is currently entering its experimental phase when the application is being adopted in a real-life scenario, in collaboration with the Soprintendenza archeologica of Pompei. Several ICT research projects all over Europe are dedicated to cultural heritage sites [2–5]. However, HyperGuide features many aspects of novelty. First, its multimedia repository is based on the semantic Web specifications. Second, its location system works well both indoor and outdoor. Furthermore, vocal technologies are exploited to make the user’s interaction with the system more natural and content presentation is adapted to the end-device. Finally, the adoption of open source software and of standard protocols and reccomendations has been a key factor in the design and development. The paper is organized as follows. Section 1 (this chapter) gives a brief introduction. Section 2 presents the overall system architecture. In section 3, our approach to the multimedia digital repository is explained. The location and tracking system is described in section 4. Section 5 is about content presentation in HyperGuide, with speech technologies and device independence being key drivers. Conclusions are drawn in section 6 where future work and directions are also mentioned.

2. HyperGuide system architecture

A logical view of the HyperGuide application and its main interactions is given in figure 1. The application main logical components are distinguishable: a three-tired, J2EE-compliant Web application; a Geographic Information System (GIS) for maps visualization and user's position rendering; a multimedia digital library for content store and retrieval; a location and tracking system; a voice engine with a text-to-speech (TTS) system and an Automatic Speech Recognition (ASR) system. Given its multichannel nature, we represent HyperGuide different access mediums (Internet, WLAN, UMTS, GPRS, etc.) and techniques with an access gateway. The diagram also represents two different types of users: the remote user and the on-site user. The former mainly logs on to the system through a desktop/laptop PC with a standard browser and can surf the Web application viewing content like pictures, video, and of course text. Audio files are also available with the proper browser plug-in. Remote PC users with a soft-phone may also exploit the vocal interaction. On-site users present static as well as dynamic characteristics (position, interest in some place or theme, route within the site, etc.). Their position can be determined with high accuracy and their mobile devices' capabilities can be described in a standard way, thus resulting in a richer user's context and ultimately in a better user experience.

To some extent, HyperGuide is a Web application implementing a virtual visit to an archeological site. For our demonstrator, we chose the archeological site of Pompei because of its richness, popularity, and complexity of experimental scenarios, but the application may easily be adapted to other sites. When a user accesses the system through a standard browser, a personal assistant guides him/her through a lightweight registration process which allows him/her to express his/her preferences in terms of language, education, age, and level of details for the visit. Other options like fruition mode (push or pull) and type of tour (e.g. generic tour, religious tour, etc.) are also selected at this time. However, if the user is visiting the site and his/her device fulfils some hardware requirements, a new scenario unfolds. A thin software program on the mobile station sends the system sensing information for it to determine the device's location. Simultaneously, another software module provides the system with a standard description of the device's capabilities, including hardware com-

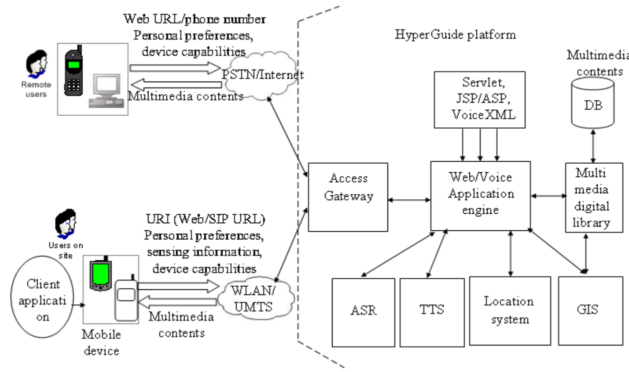


Figure 1. conceptual view of the HyperGuide application

ponents for the location determination (e.g. wireless/GPS receiver). Hence, the user's context is fully developed and the application engine may query the location system for the user's location, the digital library for the relevant multimedia content and the GIS system for the user's rendering on the site map. Before content can be presented back to the user, content adaptation also occurs. From a static, generic tour, the visit is transformed to a dynamic, customized one where the user can freely move around the cultural heritage site and receive the pertinent information. Users unable or simply unwilling to look at a device's screen and read on it may resort to the vocal interaction, which is handled by a voice over IP (VoIP)-based, bilingual speech system including an ASR and a TTS. Here, a vocal dialog is started, which first enables the user to make his/her choices of language and level of detail. The user can then move around the site and ask the virtual guide for more information. As a result, the user is presented with a list of nearby cultural resources from which to choose by just uttering their name or part of their name. Again, the recognition engine will understand the user's choice and select the content from the digital library much in the same way as a Web server does for a Web browser client. Finally, all the available information on the relevant resource, characterized by the previously selected language and level of detail, is provided to the user in the form of synthesized speech. The vocal system supports advanced features like barge-in, which allows to listen to the client utterances while still synthesizing speech. Hence, the user may interrupt the system at any time, for instance because he/she want to move on to another cultural resource in the list, by simply saying the right keyword.

3. Multimedia digital library

Our repository of multimedia contents is the Multimedia Digital Library (MDL). Instead of the traditional database-like approach to data management, we chose the semantic Web paradigm [6]. Multimedia data is stored and accessed through an Ontology Storage and Querying System (OSQS). This has the multiple advantages of semantic expressiveness, effective data querying and retrieval, and increased flexibility in that heterogeneous storing systems may be supported more naturally. In addition, this architecture allows the MDL to smoothly evolve towards a Web service model. Our first objective was to design a new ontology for the conceptualization of our domain of interest. We selected several knowledge domains: the user visiting a cultural heritage site, the user's electronic device, the cultural resources (artifacts, manufacts, buildings, wall paintings, etc.) contained in the site, the multimedia contents (images, audio, video, etc.) associated to the cultural resources and presented to the user's device, the themes (religion, buildings, food, etc.) to which cultural resources can be related, and finally the cartographic data, which indicate with great accuracy the position of cultural resources in the site. The next step was to create an ontology for each domain. Therefore, we determined the relations among entities and we associated a metadata standard to each domain. MPEG-7 [7] is particularly suitable to represent multimedia data related to cultural resources [8]. Dublin Core is also fit to cultural resources representation [9], and its open and property-centric features give it a high degree of flexibility. Hence, we chose MPEG-7 for multimedia data, Dublin Core application profiles for data about cultural heritage, user, and thematic tours, and GML for cartographic data [10]. The integrated ontology was obtained by adding custom properties and joining the relevant concepts of each ontology. Open source platforms with Java API were considered to select a suitable OSQS. We experimented with both the Sesame framework [11] (rel. 0.9.x) and related Sebor reasoner [12], and the Jena framework [13] (rel 2.1). The latter has the big advantage of supporting OWL extensions [14] and custom defined inference rules, thus dramatically extending the knowledge base. Following releases of Sesame, enhanced with a reasoner engine, are under evaluation. In fact we realized prototypes of the MDL with both platforms. Multimedia data, of different formats and sizes, were stored on a file system and described through RDF [15], while the ontology defined above was represented in RDFS [16]. The query engine of the prototypes could access and retrieve multimedia data. Both the OSQSs adopted have shown good performance and high stability and reliability. Finally, we conducted some simple performance tests, based on a common RDF knowledge base. The results showed that the RDQL query engine was faster than the RQL one on the Sesame platform. They also showed a better performance for the Sesame RDQL query engine than for Jena's.

3.1. Integration between MDL and GIS

In HyperGuide, the main goal of the GIS is to provide the user with geographically enriched context-aware information about cultural resources. A map may be rendered on the user's device screen with selectable points of interests, and the relevant information may be obtained. In addition, the GIS exports functionalities to obtain the URIs (Uniform Resource Identifier) [17] of all the cultural resources in a geographic surround of the user, for instance a circle or a rectangle centered in the user current location, as provided by the location system. We adopted a freeware GIS, namely ALOVMap [18], because of its support of standard formats (e.g. shapefile) and because it is compliant with the Web Map Service (WMS) [19] specifications. WMS allows access to maps and other GIS functionalities from any device, including mobile ones, through the HTTP [20] protocol. Moreover, it is very useful to assign some appropriate significance to points and curves drawn on the map, e.g. a line may represent the boundary of a building or its access doorway. Likewise, a cultural resource may be related to one or another particular theme or tour. To capture the variety of aspects and the semantic significance of our knowledge domain, we represented spatial data in a standard XML-based encoding called Geography Markup Language (GML) [21], introduced by the Open GIS Consortium (OGC) [22]. Then we converted those data to RDF for succeeding storing to the DML. As a result, it was possible to load geographical information to the DML and make that information available to other applications, e.g. the virtual visit. Data stored in the DML was also made accessible through the GIS interface with its associated tools. The tight integration of the DML with the GIS has several advantages: first, spatial data are given semantic significance while still keeping the accuracy and precision typical of a GIS. Second, it is possible to extend that significance by making inferences or even by defining custom inference rules. Third, all data in our knowledge domain present the same common interface since they are stored in the same repository. Finallt, the adoption of GML guarantees flexibility, interoperability among heterogeneous systems, and independence from presentation.

4. Location system

The location system provides location and tracking of the user's device both indoor and outdoor. The system has been designed from start to support any number of sensing technologies. In fact it currently leverages on two different technologies to get the best available location estimate: the Global Positioning System (GPS), which is available in external environments, and radiofrequency, in the form of an IEEE 802.11b [23] wireless local area network (WLAN) infrastructure. A conceptual view of the location system is provided in figure 2. The mobile stations communicate with a location server by sending it the relevant sensing information. This may be an NMEA [24] string from the on-board GPS receiver, or the signal strength read at the 802.11b receiver. Either case, the location server, which is technology-neutral, does not elaborate the information itself, but instead it forwards it to the specific technology-aware server responsible for the actual position determination, e.g. a GPS location server or a WLAN location server. The position estimate, represented in a standard format (e.g. latitude and longitude), is sent back to the location server, which keeps track of each mobile device's. For some devices (e.g. those equipped with both receivers), several position estimates may be available, in which case the location server selects the best one. In this way, any LCS client, willing to know the position of a mobile station, may query the location server with the device's MAC address as input. The Service Location Protocol (SLP) [25] is used in our system as a lightweight discovery mechanism for retrieving the location server transport protocol address (IP address and TCP port). The adoption of SLP gives the application a great flexibility, since no hardcoded information or static configuration files are required on the mobile stations or anywhere else. Furthermore, the distributed nature of the system's architecture improves scalability and performance.

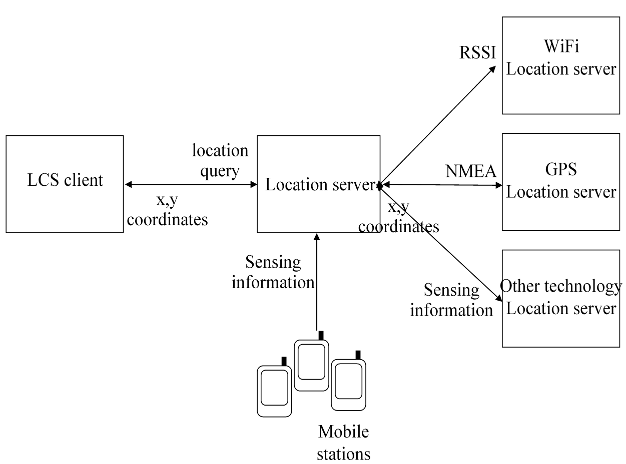


Figure 2. logical view of the location system

4.1. Components of the location system

The main components of the location system are: the GPS location server, the WLAN location server, the location agent, and the LCS client. Let's see each of them in more detail. The GPS location server receives NMEA strings from a mobile device equipped with a GPS receiver, it makes consistency checks, and extracts the device's absolute position as a tuple latitude and longitude, together with the exact time of the fix. The server then uploads this information to the location server. The WLAN location server converts real-time Received Signal Strength Information (RSSI) coming from several access points, to the receiving device's absolute position estimate. This estimate is then sent to the location server, in the same way the GPS server does. The location agent is a lightweight software program running on the mobile station, whose main task is to read sensing information from the on-board GPS/WLAN receiver(s) and to transmit it to the previously discovered location server. Finally, the LCS client is the entity concerned with a device's position. Through SLP, it discovers the location server and obtains the absolute coordinates of the input mobile device.

4.2. Experimental results with the WLAN Location system.

In our work we investigated several techniques, which we call filters, to determine the user's position inside a building with a IEEE 802.11b wireless network infrastructure. We designed, implemented and tested several such filters. We first built a database of received signal strength (RSSI) which we call fingerprints map or simply map. In addition, we implemented the well-known K-Nearest Neighbour algorithm in the signal strength space [26]. For $K >= 2$, this algorithm computes the barycenter of the K-Nearest points belonging to the fingerprints map, where "near" is associated with a particular metric. We chose the Euclidean distance but other metrics may also apply. In its simplest form, our system works as below. RSSI information from a mobile device is sent to a server which is also given the fingerprints map as input. The server then estimates the device's location using the K-Nearest Neighbour (KNN) algorithm with K set to a known value. The system can freely combine more filters. For instance it may use the circle filter and/or the signal strength filter to achieve a better accuracy. The latter focuses on the signal strength spectrum by comparing the RSSI spectrum in the map and the one in the real-time sample. The comparison is limited to a subset of cardinality M of access points, hence M is a parameter of this algorithm. This filter only considers those points in the map for which the M highest and the M lowest RSSI values are relevant to the same access points as the M highest and the M lowest RSSIs in the real-time sample. In other words, for $M = 2$ let us suppose that AP1 and AP2 are the access points with the two highest signal strength values for the real time sample (order is unmeaningful here), while AP5 and AP6 are the ones with the two lowest values. This filter selects the locations in the map where AP1 and AP2 have the two highest signal strength values and AP5 and AP6 the two lowest values. The circle filter is a software implementation of the triangulation technique using the RF signal propagation model in a vacuum. For each access point the algorithm draws a circle whose radius depends on the RSSI sample and whose center is at the access points itself. A system of equations representing the circles is solved and a shared area is output. If no solution is found, then some kind of heuristics is used until a solution is obtained. To determine the device's location we may use any triangulation algorithm but this is where the filters concept comes into play again. Starting from the shared area, we use the fingerprints map to filter out points outside the map. Finally the K-Nearest Neighbour filter is exploited to estimate the device's position. In order to validate our approach, we conducted some experiments at the ground floor of the CRIAI building in Portici, Naples. The building is 28.80 metres long by 25.20 metres wide and is divided in several different-sized offices and corridors with a large open space in the center. We first installed six wireless access points. For our experiments we used a Toshiba e800 WiFi PDA (Personal Digital Assistant) with the PocketPc 2003 operating system. We finally implemented a program that would read RSSI values on the PDA and send them to the server where the location estimation algorithm was running. With this setup, we built a RSSI fingerprints map with pace 1.80 metres. We then ran tests using these maps and the same K-Nearest Neighbour filter and the signal strength filter with M = 2. We considered 16 locations equally distributed throughout the building. To assess the location accuracy, we computed the average error distance from the actual position for each of the 16 locations. The results hinted that the signal strength filter may improve accuracy substantially. In fact, the application of the KNN algorithm alone yielded overall average distance error of 1.88 metres and a standard deviation of 2.08 metres. Applying the signal strength filter resulted in an average distance error of 1.68 metres and in a standard deviation of 1.93 metres.

5. Content presentation

In HyperGuide context is a multi-dimensional entity which refers to three aspects: the user's profile, the user's device capabilities, and the user's real-time geographical position. Once the context is acquired, and the appropriate multimedia content is selected, content adaptation occurs. This chapter is devoted to describing how and to which extent this happens.

5.1. Device independence

The driver to our content adaptation approach is the paradigm of device independence as proposed by the W3C CC/PP (Composite Capabilities / Preferences Profile) working group [27]. That provides an RDF framework for the extensible description of a device's delivery context. This is often referred to as a CC/PP profile. Both static and dynamic characteristics are considered. In this project we introduced a new CC/PP-compliant vocabulary which we called ArcheoLocation. Here, in addition to standard device capabilities, we introduced two new dynamic attributes: GPSCapable and WiFiCapable. If GPSCapable equals zero, then the device does not have a GPS receiver or its GPS receiver is switched off, but if GPSCapable equals one, then the receiver is turned on and it is acquiring a valid fix. Likewise, if WiFiCapable equals zero, then the device does not have a WLAN receiver or its WLAN receiver is turned off, but if WiFiCapable equals one, then the device's WLAN receiver is working. We implemented a C++ dynamically linked library which exports functionalities to read from the GPS/WLAN receivers on a PocketPC PDA. On the server side, we adopted the Deli [28] library, which can parse and store CC/PP-compliant profiles. We successfully tested the above configuration with an iPAQ 5550 PocketPC PDA running a Java Virtual Machine (JVM): the device sent an RDF description of the ArcheoLocation profile over the CC/PP-exchange protocol [29], and the server was able to capture it.

5.2. Content adaptation

In accordance with the W3C activity on content presentation, we used XML to collect and describe dynamic content and XSL (eXtensible Stylesheet Language) and XSLT (XSL Transformations) [30] to transform and to present that content. Let's see in more detail what happens when an HyperGuide on-site user accesses the Web application through a standard browser on a mobile device. The personal assistant has helped him/her make the right choices for him/her, for instance an expert level of detail, the English language and the religious tour. Let's assume the user is close to the cultural resource X, a temple, and to Y, a villa. The application selects expert content related to resource X only, and discards the other. Non-English texts are also filtered out. The application generates a XML document including this information, which a XSL file transforms for rendering on the user's device. Long texts, which will not fit to the small device's screen, will be split in several parts, while pictures will be properly rendered.

5.3. The speech system

In our project we deployed a bilingual speech system powered by the Loquendo ASR and TTS technology. We chose the VoIP technology instead of a more traditional telephony-like approach because of its flexibility, simplicity of integration with other Web applications, and scalability. We exploited the Loquendo system as a black box with the VoiceXML 2.0 [31] interpreter being the API layer. Grammars for the speech recognizer were developed in JSGF [32]. We implemented several vocal services. In particular, the vocalOnSite service showed the tight integration of the speech system with the digital library, with the GIS system and with the location system. Cultural resources in the user's proximity only were selected so that context-aware, multimedia data could be retrieved and then synthesized to the user. A site map including the real-time user's position was also shown to the user' s device. To make the user-system interaction more natural, the technique called barge-in was also exploited and the prosody level was fine tuned. Since a high degree of dynamicity was required, the vocalOnSite service was accurately designed so that the voice dialog and the grammar files were generated on the fly, based on the system status. Dynamic grammar files have the advantage of increased performance, higher confidence level and, as a result, they enable slimmer dialogues since there is no need to ask the user for confirmation.

6. Conclusions and future work

This paper provided a description of HyperGuide, a semantically interoperable application supporting context-aware, multimedia applications for the fruition of cultural heritage. Although work is still in progress, the project has already achieved important milestones, like the introduction of a multimedia digital library based on the semantic Web paradigm, a prototype location and tracking system for indoor and outdoor, a demonstrator of vocal interactions. We were especially pleased with the results obtained with the WLAN location system, since they demonstrate the validity of our approach. We expect more work ahead. As far as the digital library is concerned, we are experimenting on custom defined inference rules within the Jena framework. With regard to the location system, we are working on extending the WLAN system's operations to a limited range around a building with a wireless network infrastructure. We are finally planning to realize a queryBuilder, that is a module for dynamic proposition of context-aware thematic tours.

References

[1] The Italian Education, University and Research Ministry. http://www.miur.it/.

[2] UE Fifth Framework Programme. http://www.cordis.lu/ist/ka3/digicult/projects_ac.htm.

[3] UE Sixth Framework Programme. http://www.cordis.lu/ist/directorate_e/digicult/projects_fp6.htm.

[4] Archeoguide. http://archeoguide.intranet.gr.

[5] Agamennon. http://services.txt.it/agamemnon/.

[6] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, May 2001.

[7] J. M. Martnez. MPEG-7: Overview of MPEG-7 Description Tools. *IEEE Computer Society*, September 2002.

[8] J. Hunter. Combining the CIDOC CRM and MPEG-7 to Describe Multimedia in Museums. In *Museums on the Web 2002*, Boston, USA, April 2002.

[9] M. Doerr. Mapping of the Dublin Core Metadata Element Set to the CIDOC CRM. Technical report forth-ics/tr-274, CIDOC, July 2000.

[10] The Open Geospatial Consortium. http://www.opengeospatial.org.

[11] openRDF. http://www.openrdf.org/index.jsp.

[12] BOR - a Pragmatic DAML+OIL Reasoner. http://www.ontotext.com/bor/index.html.

[13] Jena. http://jena.sourceforge.net/documentation.html.

[14] S. Bechhofer and al. Owl web ontology language reference. Technical report, W3C, February 2004.

[15] RDF Primer. Recommendation, February 2004.

[16] D. Brickley and R. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. Technical report, W3C, February 2004.

[17] T. Berners-Lee and al. Uniform Resource Identifiers (URI): Generic Syntax. RFC 2396, August 1998.

[18] ALOV Map. http://www.alov.org.

[19] Web Map Service. OGC Specifications, August 2004.

[20] R. Fielding and al. Hypertext Transfer Protocol – HTTP/1.1. RFC 2616, June 1999.

[21] OpenGIS Geography Markup Language (GML) Implementation Specification. OGC Specifications, January 2003.

[22] Open Geospatial Consortium.

[23] The Institute of Electrical and Electronics Engineers, Inc. IEEE Standard 802.11 - Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, 1999.

[24] The National Marine Electronics Association. http://www.nmea.org/.

[25] E. Guttman and al. Service Location Protocol, Version 2. RFC 2608, June 1999.

[26] P. Bahl and V. Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. *IEEE CS Press*, pages 775–784, August 2000.

[27] http://www.w3.org/Mobile/CCPP/. CC/PP Information.

[28] M. H. Butler. DELI: A DElivery context LIbrary for CC/PP and UAProf. External technical report hpl-2001-260, HP, February 2002.

[29] H. Ohto and J. Hjelm. CC/PP exchange protocol based on HTTP Extension Framework. W3C Note, January 1999.

[30] J. Clark. XSL Transformations (XSLT) Version 1.0. Technical report, W3C, November 1999.

[31] Voice Extensible Markup Language (VoiceXML) Version 2.0. W3C Recommendation, March 2004.

[32] JSpeech Grammar Format. W3C Note, June 2000.

A 3D INTERACTION METAPHOR FOR REMOTE CONTROL OF SMART HOME SYSTEMS

Gennaro Costagliola, Sergio Di Martino, Filomena Ferrucci, Genoveffa Tortora
Universitàdi Salerno –Dipartimento di Matematica e Informatica, via Ponte don Melillo, Fisciano, Italy
Email: {gcostagliola | sdimartino | fferrucci | tortora}@unisa.it

Keywords: Distributed Virtual Environment, Multimedia human-computer interaction, Mobile/Pervasive Computing

Abstract

Smart home technologies are an open and attractive research field, due to the number of advantages they can provide to the every-day life. To ensure effectiveness of smart home systems it is necessary to provide users with appropriate user interfaces.

In this paper, we present a 3D multimodal interaction paradigm, specifically conceived for remote control of domestic appliances. In particular, the proposal addresses the main issues related to the interaction with domestic virtual scenarios. As a matter of fact, it provides some features which exploit different sensorial channels to support users in the navigation of the virtual home. Moreover, to support interaction with appliances it has been defined an intermediary, standard interface with which the user interacts as a remote control. The preliminary evaluations performed provide encouraging results on the effectiveness of the proposal

1. INTRODUCTION

In the literature there are dozens of descriptions about the "House of the future", and the provided services, which, exploiting the pervasiveness of sensors and actuators, will range from lights, climate and music that automatically adjust as soon as a family member enter in a room, to phones ringing only in rooms where the addressee is present, till complete security systems, able to forward emergency alarms in case of burglars, fire, or injury. As an example, Meyer provides an interesting survey about futuristic smart home scenarios [8].

Indeed, the application of computer technologies to domestic appliances is a challenging and fascinating research field. Nowadays, many appliances allow some degree of remote control. *Smart home technology* aims to integrate and extend this throughout the house, making the electronic devices to act "smart" or more automated. Indeed, by exploiting home networks and the Internet, smart home technologies will allow the appliances to communicate with each other and to react to stimuli, (i.e. to be context-aware), and in the meantime the homeowners to control domestic systems in any part of their house, while being eventually remote.

Two main categories of users can specially benefit from those systems: people who have the need to remotely control their appliances from great distances, such as controlling a device at home from the workplace, and people with disabilities, having limitations in movements.

Obviously, the development of novel, safer and effective *smart home systems*, involving the automaton and the (eventually remote) control of appliances, is posing demanding challenges to *Information and Communication Technologies* specialists, ranging from security to robotics, from context-awareness to Human-Computer Interaction [8]. Indeed, the enhancements in the number of features offered by smart homes should be paired by equivalent improvements in the development of comprehensive, comfortable and self-explaining user interfaces [2]. Currently WIMP (Windows-Icon-Mouse-Pointer) is still the most diffused interaction paradigm, even if some researches provide evidence that an effective interface for smart homes should provide a clear and intuitive imagination about the home environment, (i.e. a geometrical view) [3]. As a result, Virtual Reality approaches can provide increased access, convenience, and efficiency in smart home control [2], both by remote homeowners, and disabled people.

In this paper, we present a 3D multimodal interaction paradigm specifically conceived for remote control of domestic appliances. The proposal addresses the main issues of interacting with virtual scenarios, providing effective solutions to support the navigation and the interaction with the domestic appliances. In particular, the key idea is to apply the concepts underlying the *Personal Universal Controller* (PUC) [4] to the virtual environment, i.e. to design an intermediary, standard, interface with which the user interacts as a remote control for any appliance.

Moreover, the auditory channel is exploited to enhance the context awareness, and to improve the navigation within the virtual scenario.

The paper is structured as follows: in section 2 we present our multimodal interaction paradigm, detailing the facilities provided to support the navigation and the interaction with the appliances. An example and some preliminary evaluation results are also provided. In section 3 we briefly recall the main related researches. Some final remarks and a discussion on future work will conclude the paper.

2. THE INTERACTION PARADIGM

The availability of high Internet bandwidth and powerful 3D graphic hardware for a broad range of personal computer users has generated a strong momentum of convergence between these even faster and cheaper technologies, aimed to bring appealing 3D graphics on users' desktops via Internet. As a result, many fields of the Information Technology have been affected by the charm of the 3D, due to its many intrinsic advantages.

In particular, in the field of smart homes, virtual reality can provide user with the advantages of a "real" view that is more effective than metaphors of WIMP objects. For instance, (s)he can get a realistic view of the house, with its structural components like rooms, appliances, etc…, displayed with correct spatial localization and context. Moreover, user can directly manipulate appliances, like turning lights on or off, and get immediate visual feedbacks. This is much more effective than a textual multi-level hierarchy, where appliance status is depicted with a label like "room 3 – light 2 – On" [3].

Summarizing, when dealing with smart homes, virtual reality has a number of inherent appealing properties, namely:

- Visual awareness about appliances status.
- Improved situational and spatial awareness, if compared with WIMP interfaces
- Real-time interaction
- Multi-modality
- Sense of presence
- Ability to change user's scale in time and space

However, in spite of these opportunities offered by the 3D paradigm, there are still some drawbacks that prevent a full exploitation of the potential of interfaces based on 3D [13]. In particular, we can outline three main difficulties when moving in the virtual domestic environment:

1. Navigating the scenario, to reach an intended location of the house
2. Locate the object (typically the domestic appliance) to interact with, and
3. Interact with the object.

Our paradigm was conceived to simplify all these three tasks. In the following we are providing more details. In particular, we describe in section 2.1 the provided support on the navigation task, in section 2.2 how the paradigm supports the localization of the intended appliances, and in section 2.3 how to interact with domestic devices. An example in section 2.4 clarifies the proposed paradigm.

2.1 Navigation through the smart home

In order to reach a specific location of the home, user has to "navigate" through the scenario. Researches on interaction with virtual environments reports that 3D navigation can be divided into two components: the motor one, named *travel*, and the cognitive one, named *wayfinding* [1]. The former task involves the movement of the viewpoint from a location to another. The latter one is "the cognitive process of defining a path through an environment, thereby using and acquiring spatial knowledge to build up a cognitive map of an environment" [1].

Since it is widely recognized that current Web3D browsers do not offer satisfactory support for the navigation task ([13],[16]), we enhanced the domestic scenario with some multimodal interaction widgets, intended as aids for the user travel tasks. Indeed, in the smart-home domain, travel could be intended both in the service of some other task (user wants to reach a specific appliance), both as an end unto itself task (user walks in the home, looking for appliances settings, such as lights, curtains, etc…).

In the first case, to save user's time/efforts, the interface provides a kind of tele-transport feature: user gets a map of the house, and can point & click on the intended room to be virtually transferred at its entrance. Optionally, (s)he can also select in a list an appliance for the specific room, to be placed directly in front of it. A screenshot of such a feature is provided in Fig. 1.

Otherwise, if user is interested in navigating through the virtual scenario of his/her home (e.g. for control purposes), other than relying on the typical features provided by the Web3D browsers, such as Parallel Graphics Cortona [14], or Blaxxun, (s)he can exploit both a "target" widget we embedded in the scenario, which is detailed in the next section, and some multimodal features suited to improve user's spatial knowledge. In particular, some previous research on 3D environments [6] has shown that navigation in virtual scenes is simplified if users can receive different information about scene structure and orientation, simultaneously from different senses. Indeed, efficiency of navigation (wayfinding) is improved when both vision and hearing, that are all part of a multimodal experience, are involved. Researches show that this apparent redundancy is often useful in many situations of interaction with virtual scenarios [13].

Fig. 1: Navigation (travel) support

Starting from these bases, we exploited the concept of *Interaction Locus* [13]. With this approach, zones of the scenario with some inherent relations are grouped in themes, named Interaction Loci. Each theme is characterized by homogeneous morphologic features, has the same interaction modalities and informs the user about its specific nature by starting a coordinated set of information streams each time the user enters inside of it. Such information streams involve several user senses, e.g. using parallel visual and auditory communication channels. Thus, multimodality, a peculiarity of the real world experience, is used to overcome the limits of vision in desktop virtual environments, allowing the user to gain more awareness of the 3D world structure for wayfinding purposes [13].

This approach was successfully applied in many domains, with particularly relevant results in the filed of cultural heritage and 3D virtual museums [15].

In the smart home context, it is quite usual to identify different zones of the house, and thus the concept of Interaction Locus fit very well to improve wayfinding. In particular, in our application, this concept was implemented by defining different auditory backgrounds that are associated to different section of the house.

2.2 Appliance localization

In order to support the user in starting the interaction with domestic applications, we defined two kinds of metaphors, intended to:

1. facilitate the user travelling towards appliances, and
2. activate the specific interface for interacting with the appliance.

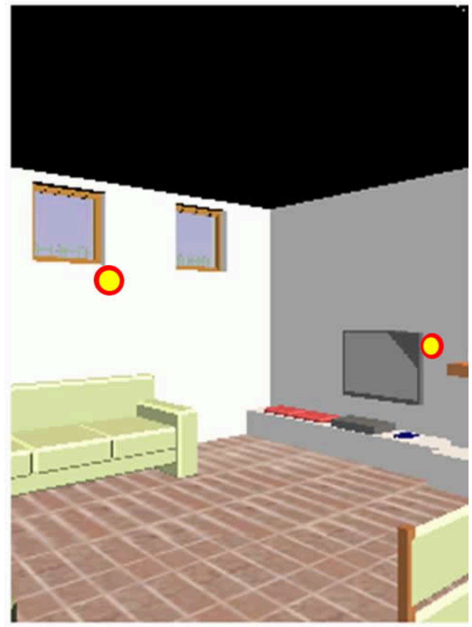


Fig. 2: Widgets supporting the identification of active objects

Fig. 3: Labels to identify and interact with appliances

The former facility is suited to avoid that user has to manually move in the 3D space to reach a specific appliance with which interact. Indeed, near active domestic devices, the interface provides a big red-yellow target, visible from distance. Such target can be clicked. In this case, 3D browser automatically put the user's viewpoint in front of the intended appliance. An example of the targets is provided in Fig. 2

The latter facility is represented by a green label indicating the name of the appliance. If clicked, the label activates the remote control (described in the following section). Such labels are displayed only when the user is near enough to get a comfortable reading of the text (for instance, after having clicked the target). An example is provided in Fig. 3

2.3 Interaction with appliances

As standard metaphor to mediate interaction with the appliances, we designed a kind of Universal Remote Control (URC). The key idea is to provide the user with a standard, semitransparent interaction panel, which pop-ups over the scene.

Such panel is composed of two set of widgets: some of them are identical throughout all the appliances, to provide the user with a standard interface, while some others are specific for the intended device.

Moreover, if a device requires special controls, a dedicated area of the URC is intended to show a further panel with the other controls. The choice to hide some of them has a twofold motivation. The first is to not visually /cognitive overload the user with non-frequently selected controls, showing only the main ones, while the second is to minimize the display area covered by the URC.

An example of such an URC is provided in Fig. 4, where it is depicted the Tuner control.

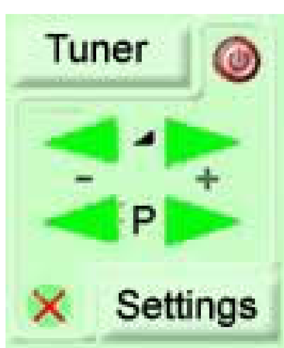


Fig. 4: An example of Universal Remote Control

By analyzing the layout of the URC, it is possible to notice that it is horizontally divided in three zones:

1. An upper area, encompassing a label indicating the name of the controlled equipment, and a red button, suited to put ON/OFF the equipment. This area is the same for all the devices.
2. A central section, embedding the appliance-specific controls. They are usually intended as pairs, and are aimed at controlling settings and/or features specific for the device. Recalling Fig. 4, with the tuner, we have the faculty to adjust the volume, or the tuned station.
3. A lower area, encompassing two buttons: the first (a red X) is suited to hide the URC, while the second is intended to active a device-specific panel. Also this area is the same for all the devices.

2.4 An example of interaction

In Fig. 5 it is presented an example of interaction based on the above depicted paradigm.

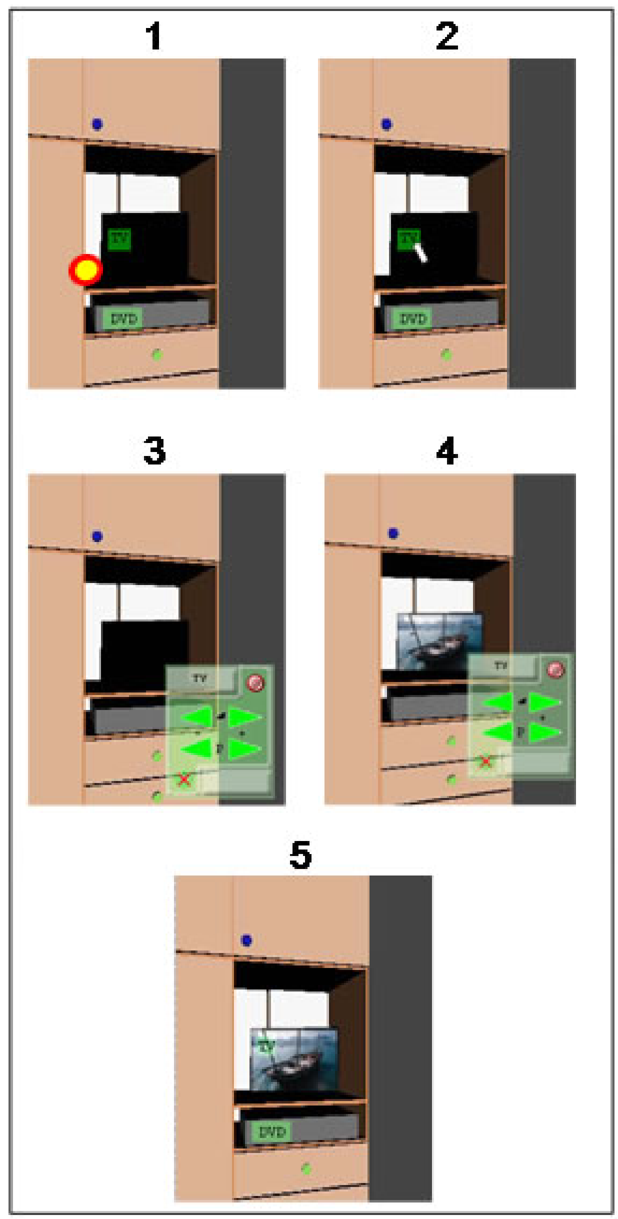


Fig. 5: An example of interaction

For instance, let us suppose a disabled user wish to put on the TV color (or a parent wants to remotely control the TV program the son is looking).

After selected the suited room with the map in Fig. 1, user is automatically "teleported" into such space. Then, the user clicks with the mouse pointer on the red target, and the 3D browser moves his/her viewpoint smoothly in front of the appliances (Fig. 5 - 1). User can have immediate feedbacks that the TV is off.

Then, to put on the device, user clicks on the label "TV" to start the interaction (Fig. 5 - 2). As a result, the system displays the URC, with the suited widgets (Fig. 5 - 3). User presses the power button in the top-right corner of the URC, and the TV starts-up (Fig. 5 - 4). Eventually, user can change the channel and/or adjust the volume with the controls on the RC. Then, to terminate the interaction, user clicks on the "X" button in the lower left corner, to close the RC (Fig. 5 - 5).

2.5 Some preliminary evaluations

We have conducted some preliminary evaluations of the proposed paradigm. To have a meaningful scenario, we developed a 3D domestic environment, representing a house with five rooms and several active appliances, such as lights, curtain, TV, DVD player, etc…

The scenario was implemented in VRML (*Virtual Reality Modeling Language*) [10], the *de facto* standard for describing interactive platform-independent 3D environments over the Web. The interaction with appliances was simulated through opportune VRML animations, video clips, etc…

The interaction with the user was based on the two alternatives offered by VRML to control the objects composing the scene: the *ROUTE* mechanism and the *External Authoring Interface* (EAI) solution. The former was mainly adopted to handle the interaction both with targets (i.e. to move the waypoint in front of appliances after the click) and with the Remote Control. The latter was used, together with a Java Applet, to create and "make live" the map shown in Fig. 1

Obviously the proposed interface can be very easily integrated with COTS smart home platforms, such as *Echelon Lonworks*.

Since it is widely recognized that evaluations cannot rest totally on designers and experts, but it is important to solicit feedbacks from the user community itself, we involved a group of eight external subjects (age ranging from 25 to 59, 4 with previous skills on 3D navigation and 4 unskilled) in the evaluation tasks.. We recall that many researches (such as [5]) report that eight is an adequate number for UI evaluation.

Users were previously trained on house structure, then they were asked to complete some tasks (such as select a channel on TV, open the windows, lock the garage door, etc…), and finally they were asked to fill in a questionnaire after they completed the trial session.

During the experiment, we recorded dependent measures, such as time to complete a task, followed path, and interaction errors.

Although the obtained results can be considered preliminary, they are very encouraging for our research. For instance, all subjects were able to successfully complete the assigned tasks with few efforts. For sake of brevity, we cannot provide all the results. However, in Table 1 we report the number of errors/indecisions during interaction for three tasks. S1-S4 are the skilled subjects, while S5-S8 are the unskilled ones.

Table 1: Error/indecisions during interactions

	TV	**Window**	**Door**
S1	0	1	0
S2	1	0	0
S3	0	0	0
S4	2	0	0
S5	2	1	2
S6	2	1	1
S7	1	1	0
S8	3	2	1
Mean	*1,4*	*0,8*	*0,6*

About the results of the questionnaire, in mean the subjects agreed with the proposed paradigm. In particular they felt very useful the target widget, sufficiently effective the RC, while some concerns remains during the free navigation of the virtual scenario.

3. RELATED WORK

As stated in the previous sections, currently very large research efforts are being devoted towards the smart home domain, and in particular on the definition of context-aware appliances.

Even limiting our focus on interaction with smart homes, a number of research groups are working on this argument, and in particular on the definition of effective and universal remote controls. In particular, many studies are dealing with equipment and functions to be controlled from handhelds. For instance, Myers is working on the concept of universal remote control [4],[7] to manage domestic applications, by defining a common interaction protocol. The Stanford *ICrafter* [11] is a framework aimed at supporting an automatic generation of interfaces, even if it is mainly focused on the infrastructure aspect. Another similar project is *XWeb* [12], whose main goal is to automatically generate interfaces with complex layouts.

About multimodality, Potamitis et al. are investigating the usefulness of vocal interaction to control remote appliances [9]. However, all these projects deal with "real" environments.

About 3D interfaces for smart homes, some researches were conducted by the University of Munich ([2],[3]), with the generation of a Java3D user interfaces, aimed at interacting with a domestic environment. However, this project was mainly focused on defining a flexible framework for representing houses using an *Object-Oriented Structure*, so it lacks of a specialized paradigm for appliance control.

As a result, at best of our knowledge, there is not a comprehensive solution, covering both aspects of navigation and interaction with virtual scenarios of smart homes.

4. CONCLUSIONS AND FUTURE WORK

Diffusion of smart homes applications is quickly rising. Nevertheless to ensure effectiveness of these systems it is necessary to provide users with appropriate user interfaces. It is recognized that traditional WIMP interfaces can turn out to be not fully adequate. Thus, there is the needing for novel interaction paradigms.

In this paper, we presented a 3D multimodal interaction paradigm, specifically conceived for remote control of domestic appliances. The proposal provides effective solutions to support the navigation (both the travel and the wayfinding components) and the interaction with the domestic appliances. The last aspect is covered by the Universal Remote Control, a pop-up panel with a standard layout, recalling the remote controls.

Preliminary evaluations performed on a sample of eight end-users provide encouraging results on the effectiveness of the proposal.

REFERENCES

[1] Bowman, D., Kruijff, E., LaViola, J., and Poupyrev, I. *An Introduction to 3D User Interface Design. Presence: Teleoperators and Virtual Environments*, vol. 10, no. 1, 2001, pp. 96-108

[2] L. Borodulkin, H. Ruser, H.-R. Tränkler: *Smart Home 3D User Interface Man-Machine Interaction with Object Oriented Approach*, International Conference Sensors and Systems, 24.-27. June 2002, St. Petersburg, Russian

[3] L. Borodulkin, H. Ruser, H.-R. Tränkler. *3D Virtual ‚Smart Home‘ User Interface* . Proc. of VIMS2002 International Symposium on Virtual and Intelligent Measurement Systems. Mt. Alyeska Resort, AK, USA, May 2002

[4] Nichols J., Myers B, *Generating Consistent Interfaces for Appliances*, in the Second Workshop on Multi-User and Ubiquitous User Interfaces (MU3I) at Intelligent User Interfaces IUI'05. January 9, 2005. San Diego, CA

[5] Shneiderman B., 1998, *Designing the User Interfaces*, Addison Wesley.

[6] Fogli D., Pittarello F., Celentano A., and Mussio P. (2003). *Context-Aware Interaction in a Mobile Environment*, Proc of Mobile HCI 2003, LNCS 2795, pp. 434-439, 2003.

[7] Nichols J., Myers B. Studying The Use of Handhelds to Control Smart Appliances. *International Workshop on Smart Appliances and Wearable Computing.* Proc of the *23rd IEEE Conference on Distributed Computing Systems Workshops*. May, 2003, Providence, Rhode Island. pp. 274-279

[8] Meyer, S. and Rakotonirainy, A. (2003). *A survey of Research on Context-Aware Homes*. Proc. of the Australasian information security workshop conference on ACSW frontiers 2003, pp. 159–168.

[9] I. Potamitis, K. Georgila, N. Fakotakis, G. Kokkinakis: *An Integrated System for Smart-Home Control of Appliances Based on Remote Speech Interaction* , EUROSPEECH 2003, 8th European Conference on Speech Communication and Technology, pp. 2197-2200, Geneva, Switzerland, Sept. 1-4, 2003

[10] VRML specification, ISO/IEC 14772-1:1997

[11] Ponnekanti, S.R., Lee, B., Fox, A., Hanrahan, P., and T.Winograd. ICrafter: *A service framework for ubiquitous computing environments*. Proc of UBICOMP 2001. pp. 56-75.

[12] Olsen Jr., D.R., Jefferies, S., Nielsen, T., Moyes, W., and Fredrickson, P. *Cross-modal Interaction using Xweb*. Proc. of UIST'00: ACM SIGGRAPH Symposium on User Interface Software and Technology. 2000. pp. 191-200.

[13] Pittarello, F.: *Accessing Information Through Multimodal 3D Environments: Towards Universal Access*, Universal Access in the Information Society Journal, 2(2) (2003) 1-16

[14] ParallelGraphics Cortona Player, available at http://www.parallelgraphics.com/cortona

[15] F. Pittarello, *Multi sensory guided tours for cultural heritage: the Palazzo Grassi experience*, International Cultural Heritage Informatics Meeting (ICHIM) 2001, Milan, Sept. 3-7 200

[16] G. Costagliola, S. Di Martino, F. Ferrucci, *A comparison of Web3D technologies from a developer's and an end-user's point of view* , in Proceedings of KSI on The Fifteenth International Conference on Software Engineering And Knowledge Engineering (SEKE '03), San Francisco, CA, USA, 2003. KSI Press.

Rapidly Prototyping Multimedia Groupware

Michael Boyle and Saul Greenberg

***Abstract*—Multimedia groupware systems provide rich support for distributed team work. Yet effective design of these systems is difficult because they must cater to complex human and social factors. Rapid prototyping partially mitigates this, for it allows designers to build, deploy, test and quickly evolve design ideas. The problem is that multimedia groupware is hard to prototype because distributed multimedia systems are complex to implement. To solve this problem, we offer the Collabrary, a toolkit designed to ease prototyping of multimedia groupware. The Collabrary blends real-time streaming multimedia, asynchronous shared application state, and novel multimedia analysis and manipulation algorithms to provide rich functionality for distributed teamwork. Implementing core functionality—multimedia capture, analysis, manipulation, transmission and rendering—is trivial. The Collabrary also affords lessons that inform the design of universally accepted toolkits for building distributed multimedia systems: we illustrate why toolkits should be accessible for learnability, lightweight so simple ideas are simple to build, and flexible so that novel unanticipated ideas are possible to implement.**

***Index Terms*—distributed multimedia groupware, prototyping.**

I. Introduction and Motivation

INCREASINGLY, groupware systems are incorporating multimedia functionality. Systems such as Instant Messaging now add pictures, voice and video to what was once a simple text channel. Scores of experimental *multimedia groupware systems* supporting distributed colleagues treat multimedia as first-order data types [1,2] blending streaming multimedia with persistent shared application state.

Yet multimedia groupware design is challenging. It must cater to complex human and social factors to support both individual and team work practices [3]. One well-known method of handling this design challenge is *prototyping*, i.e., "artifacts that simulate or animate some but not all of the features of the system" [4]. Prototypes vary in fidelity and purpose, but all lead to iterative design. A *low-fidelity* prototype might consist of ideas sketched on paper to quickly get a sense of the major design concept. A *medium-fidelity* one can be a first-cut subsystem implementation that helps one determine factors such as feature usability and/or system performance. A *high-fidelity* prototype can be an extensive interactive user interface that can be deployed to users and marketers for feedback.

 Funded by NSERC & MSR.

M. Boyle and S. Greenberg are with the Dept. of Computer Science, University of Calgary, 2500 University Drive NW, Calgary, AB T2N 1N4 Canada (phone: 403-220-6087; e-mail: {saul *or* boylem}@ cpsc.ucalgary.ca).

However, satisfying the socio-technical design problem of multimedia groupware requires *working system* prototypes: initial implementations of the system deployable to resilient users who do not mind occasional glitches and restarts. These prototypes may: be constrained to idealized hardware, software, and network platforms; be deployed only over a secured network or within benign social situations to alleviate security concerns; or contain only a subset of expected functionality. This limited deployment is extremely valuable. It helps the designer uncover socio-technical issues that are otherwise hard to detect except under extended, real use [5].

The problem is that multimedia groupware is hard to prototype because distributed multimedia systems are complex and difficult to implement. As a solution, this paper offers the Collabrary, a toolkit specifically designed to allow developers to easily prototype distributed multimedia groupware.

A. *Toolkits for Multimedia Groupware*

Greenberg [6] argues the need for easy-to-program toolkits for novel interface areas: "By removing low-level implementation burdens and supplying appropriate building blocks, toolkits give people a 'language' to think about these new interfaces, which in turn allows them to concentrate on creative designs." Yet it is hard to develop a toolkit for building distributed multimedia systems because they require a wide gamut of hardware and software infrastructure.

On the multimedia side, there must be OS support for A/V hardware, and libraries for capturing, compressing, and rendering multimedia. On the network side, multimedia must be distributed to all machines participating in the groupware session. This may require basic communication services (e.g., TCP, UDP and multicast IP), time-synchronization of multiple concurrent data streams (e.g., RTP [7]), and session management (e.g., SIP [8]). It may also require protocols for coordinating application behavior and sharing state (e.g., RPC, XML Web Services [9]), notification services (e.g., Elvin [10]), relational databases and/or distributed shared memory (e.g., JSDT [11]).

While some distributed multimedia toolkits provide robust and high-performance streaming multimedia services (e.g., [12,13]), they omit rich support for sharing other sorts of application data needed in groupware. Similarly, groupware toolkits that support application data sharing (e.g., [14]) do not robustly handle multimedia. Using these two classes of toolkits together simultaneously is awkward, as they use mutually incompatible programming environments and idioms.

Balancing abstraction and performance is tricky. A balance that is appropriate for prototyping will like not be appropriate for production, and vice versa. For example self-contained applications like Microsoft NetMeeting [15] are easy to use and offer good performance but can be remotely controlled in only limited ways. Commercial toolkits like Microsoft DirectShow [16] and JMF Java Media Framework [17] are more flexible but are also incredibly complex to learn to use.

In this paper, we present the Collabrary, a toolkit we developed to aid the rapid implementation of working system prototypes of multimedia groupware applications. It is implemented as a Microsoft COM object library and can be used with popular rapid application development platforms (e.g., Visual Basic, C#, Python) as well as lower-level languages like C++. In the sections that follow, we explain the requirements for this toolkit and illustrate how the Collabrary meets these requirements.

While the prototypes produced with the Collabrary are optimized and robust enough for limited deployments that suffice for understanding complex socio-technical factors, they are not as optimized or robust as systems made with production-oriented toolkits. The benefit of using the Collabrary is that prototypes made with it are highly malleable and quickly implemented, matching the needs of prototyping.

II. Toolkit Requirements.

The need to implement working system prototypes *rapidly* makes special demands of the toolkit used. In particular, it must *trivialize common programming tasks* so that prototypes can be built and rebuilt from scratch quickly. This allows end-programmers to focus on implementing the novel aspects of the design, and make substantial, deep revisions to the prototypes without lamenting time lost on prior unsatisfactory versions. In the Collabrary, we have sought to make common programming tasks trivial in three important ways that will be discussed extensively in the remainder of this paper.

- *Accessible*: the toolkit should be easy to learn, so that novice toolkit users can develop applications after only modest training. To meet this goal, we emphasized simple programming idioms already familiar to end-programmers.
- *Lightweight*: common tasks should need very little code to implement, using simple programming statements. To meet this goal, we designed the Collabrary to provide rich functionality that is difficult or tedious to implement from scratch. It performs many important tasks automatically or as default behavior.
- *Flexible*: the toolkit should be supple enough to design a wide range of unanticipated applications. To meet this goal, the Collabrary provides direct access to multimedia data so they can be altered, uses programming idioms borrowed from other application domains already proven flexible, and allows optional customization of default behaviors.

We also identify four common multimedia groupware programming tasks that we strive to make accessible, lightweight, and flexible in the Collabrary.

- *Capturing* multimedia must be trivial.

```
class MainForm : Form {
  PictureBox pictureBox;
  Camera camera=new CameraClass();
  Microphone mic=new MicrophoneClass();
  Speaker spkr=new SpeakerClass();
  MainForm() {
    camera.Captured+=…camera_Captured…;
    camera.Size=…320x240;
    camera.FrameRate=15;
    mic.Captured+=…mic_Captured…;
    mic.Recording=true;
  }
  void camera_Captured(IPhoto frame) {
    pictureBox.Image=…frame…;
  }
  void mic_Captured(IWaveform samples) {
    spkr.Play(samples);
  }
  [STAThread] static void Main() {
    Application.Run(new MainForm());
}}
```

Fig. 1. Capturing and rendering multimedia: `Collabrary.Camera`, `Collabrary.Microphone`, and `Collabrary.Speaker`.

- *Rendering* multimedia must be trivial and compatible with a rapid application development GUI toolkit.
- Simple multimedia *manipulation & analysis* must be trivial, while advanced manipulations must be possible.
- *Transmitting* multimedia and application state data must be trivial and match how end-programmers work with the data.

In the following sections, we describe how a Collabrary end-programmer achieves these four common programming tasks. We use snippets of C# code to illustrate the toolkit in action, and discuss important aspects of the API relevant to the task.

III. Capturing Multimedia

Perhaps the most routine multimedia toolkit program task is audio/video capture. Yet, even this can be difficult as it involves discovering capture hardware, accessing it, configuring capture properties (e.g., video frame size and rate, audio sampling rate) and then controlling capture. The Collabrary trivializes multimedia capture by offering simple hardware abstractions and by notifying the programmer of multimedia acquisition through a familiar event-based paradigm. Fig. 1 shows Collabrary program code that illustrates trivial audio/video capture.

A. Hardware Abstractions

The Collabrary provides end-programmers with succinctly-named classes that encapsulate high-level abstractions of multimedia hardware. In Fig. 1, video and audio are captured by a `Camera` and a `Microphone` object, respectively. The key is that these abstractions remove unnecessary programming complexity while adding robustness.

The `Camera` class will be used to illustrate six ways the Collabrary makes multimedia capture simple yet robust. The principles apply equally to audio and file-based multimedia input (not shown in the figure). Some principles are analogues of patterns employed in other prototyping toolkits while others emerged during the process of employing the Collabrary in various prototypes [1,22].

1) It works with any ‘plug and play’ camera; the programmer does not need to specify device-specific properties.
2) The program runs without exception even if no camera is

attached to the computer: the `Camera` object automatically inserts a 'test pattern' image in place of live video.
3) The program continues to function even if the camera is detached, and automatically connects as soon as a new camera is attached.
4) Multiple copies of the `Camera` object can simultaneously share access to the same camera device.
5) Objects require little initialization before they may be used because their properties are embedded with useful defaults. These can be overridden: the `Camera.FrameRate` default of 0 fps, which indicates manual capture, is reset to 15 fps to start automatic capture. The frame size is also specified.
6) No 'shutdown' or 'cleanup' code is required: the objects gracefully release resources when garbage collected.

These features are not implemented in other toolkits.

B. Event-Oriented Architecture

To promote accessibility, the Collabrary manages multimedia capture using the event-driven callback paradigm familiar to GUI programming. When multimedia is captured by a Collabrary object, the object "raises an event." The end-programmer can attach a callback method to handle the event.

As seen in Fig. 1, these event handlers for the video camera and microphone are attached in the `MainForm` constructor. The `camera_Captured` method handles the `Camera.Captured` event and is invoked each time a video frame is captured. The captured frame is passed as a parameter to the event handler. Audio is treated similarly, where the `mic_Captured` method handles the `Microphone.Captured` event. Periodically, after collecting a small block of audio data (by default, every 50 ms of audio) the `Microphone` object will raise its `Captured` event.

There are two main advantages of this event-based idiom. First, it uses an asynchronous programming paradigm that end-programmers will already find familiar as it uses the same event dispatch mechanisms, syntax, and programming patterns as the GUI toolkit. Second, read/write access to multimedia is provided directly as a natural consequence of handling the events. However, there is a trade-off. Stream-oriented pipeline architectures (e.g., [17]) automatically timestamp data in all streams using a common reference clock to ensure audio and video streams are tightly synchronized. While Collabrary end-programmers could implement timestamps themselves, it is not a feature of the event-oriented architecture.

IV. Rendering Multimedia

The second most important task that a multimedia groupware toolkit must support is trivial rendering of multimedia that has been captured and transmitted.

Fig. 1 shows how a programmer trivially renders the captured audio/video to the local machine's GUI display and sound hardware. video rendering with the Collabrary makes use of the image rendering classes and widgets provided by the GUI toolkit (e.g., the PictureBox class in C#). Other multimedia toolkits typically render video into a widget it provides. Often, these private widget implementations are tightly coupled to the rest of the multimedia architecture for performance reasons. Yet, the standard GUI widgets compatible with the Collabrary offer adequate performance, and using them affords three critical advantages.

First, it keeps the toolkit accessible. End-programmers do not need to learn how to use a new widget. Private widget implementations often provide APIs that are inconsistent with that of the GUI toolkit. This makes it difficult for end-programmers to get started using the multimedia toolkit.

Second, it keeps the toolkit flexible**.** New forms of user interaction with the video display via the mouse/keyboard can be implemented using conventional UI programming patterns and practices. Private widget implementations are often incomplete, and do not expose mouse or keyboard input event bindings that support new forms of user interaction.

Third, it keeps the toolkit *interoperable*. The widget used to render video with the Collabrary is assured to work in perfect harmony with the rest of the GUI toolkit and can be easily composed with the rest of the application's GUI using the visual interface designers already familiar to the end-programmer. Private rendering widgets are often implemented as a popup window that cannot be visually integrated with the rest of the application's GUI or configured with the visual interface designer. In extreme cases, the multimedia toolkit may be entirely incompatible with the GUI toolkit and impossible to use.

V. Multimedia Manipulations

The Collabrary is intended to support the rapid prototyping of *novel* multimedia groupware applications. In these kinds of applications, the designer may want to manipulate audio and video in a variety of ways. To support the rapid prototyping of novel multimedia interactions, the Collabrary must make analyzing and manipulating audio and video trivial.

A. Pre-Packaged Manipulations

Some basic manipulations are anticipated, and consequently the Collabrary offers a number of pre-packaged audio and video manipulations. One example is background subtraction and replacement. Fig. 2 shows a modification to the code in Fig. 1 to implement background subtraction/replacement with the `frame.Subtract` method. Other examples include: video filters such as pixelization, blurring, and posterizing; image composition such as alpha blending; and, raster graphics primitives.

The Collabrary also has a few built-in analysis algorithms. For example, Bradski's CAMSHIFT face-tracking algorithm [18] is implemented by a `Collabrary.FaceTracker` object. With this object, the position and size (in pixels) of a face in the video can be obtained with the addition of a few simple method calls. As another example, a motion-detection

```
void camera_Captured(IPhoto frame) {
  Photo newbkg=new CameraClass();
  newbkg.Load("newbkg.jpg");
  frame.Subtract(frame, newbkg, …);
  pictureBox.Image=…frame…;
}
```

Fig. 2—Background subtraction and replacement is trivial in the Collabrary.

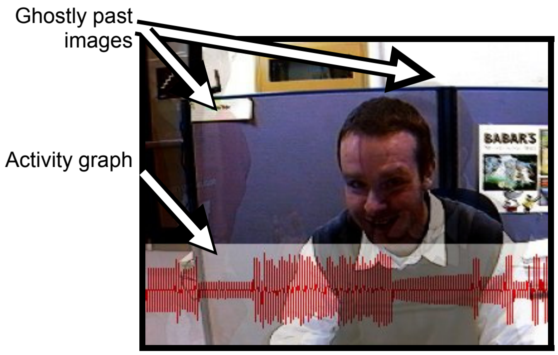


Fig. 3. Visual and graphical traces of activity, implemented by composing various pre-packaged Collabrary manipulations.

algorithm can be prototyped in just a few lines of code that use image subtraction without background replacement and compare the `Photo.PSNR` (peak signal-to-noise ratio) value of the delta image against a threshold.

B. Composing Effects from Pre-Packaged Manipulations

Custom effects can be easily achieved by composing several manipulations and analyses together. For example, Fig. 3 illustrates a sophisticated custom video manipulation, inspired by [19], built by composing the pre-packaged analysis and manipulation algorithms provided by the `Collabrary.Photo` object. Just 30 lines of code completely implement what is seen. Low-frame rate video snapshots are visually blended together to show a history of activity i.e., a frame is alpha-blended to the history of recent video frames only when it differs markedly from the previous snapshot in the history. Thus we see 'ghostly' versions of the person in Fig. 3 as he has moved about. Also, a scrolling EKG-like diagram appears at the bottom of the video. This diagram represents the activity level in the video over time. The motion detection scheme mentioned previously is used to detect changes, and the `Photo.DrawLine` method is used to draw the chart lines.

C. Custom Direct Read/Write Manipulations

The Collabrary multimedia data types (`Photo` for video frames and `Waveform` for audio sample blocks) provide end-programmers with direct read/write access to the buffered data. Two types of access are provided (not demonstrated due to space constraints). One type provides 'safe' high-level (but only modestly efficient) methods to read and write pixels and audio samples as though they were in a 2D array. The other type is 'unsafe' but highly efficient access, where the end-programmer acquires a pointer to the underlying data buffer in memory. This pointer can be passed to high-performance implementations of very sophisticated analysis and manipulation algorithms. This allows end-programmers the opportunity to quickly prototype or reuse a broad spectrum of effects and measures.

D. Event-Oriented Architecture Eases Manipulation Tasks

The event-oriented capture pipeline architecture used in the Collabrary makes implementing multimedia manipulations much more accessible and lightweight compared to toolkits based on stream-oriented architectures. With the Collabrary, the end-programmer adds manipulation code to an existing capture event handler at a minimum cost of one additional line of code. Conversely, with a stream-oriented architecture as in the JMF the end-programmer must write a filter class and insert an instance of it into the pipeline at an appropriate place. The problem is filters are difficult to write. They implement extremely generalized interfaces which treat audio and video as generic byte arrays rather than rich image or audio types. Ultimately, the programmer is forced to implement mundane code—well in excess of just one line—that is irrelevant to the real work of the filter. This heavy "up-front" work does not match the needs of rapid prototyping.

VI. TRANSMITTING MULTIMEDIA

Lastly, the Collabrary makes distributing multimedia data across networks to other computers trivial. Implementing this task makes use of: session management protocols; audio/video codecs (e.g., [20]); transport protocols that account for late, lost or out-of-order messages; and, protocols for negotiating, monitoring and regulating quality-of-service (QoS).

This task is the most difficult to implement robustly, but is essential for deployable groupware. First, the algorithms and protocols themselves are conceptually complex. Second, implementations must be carefully coded to meet performance requirements and robustly handle a myriad of possible exceptions. While there are many toolkits to insulate the end-programmer from the gory details of implementing standards robustly, they often require:

- set up/administration of network services that are separate software downloads e.g., SIP requires proxy servers;
- network features unavailable to intended prototype users, e.g., OpenMash [12] requires multicast IP; or,
- multiple toolkits to be used concurrently e.g., RTP does not provide a guaranteed lossless in-order delivery stream for arbitrary-length messages, making it inappropriate for sharing certain kinds of application state information.

The Collabrary does not implement popular Internet protocols like SIP and RTP because some of the programming idioms used with them are not trivial enough for rapid prototyping. Simpler-to-program, but less robust and efficient protocols are provided, instead.

Fig. 4 shows code that implements a simple *n*-way videoconferencing application. For brevity, audio support has been omitted, but if included it would follow similar programming patterns as video. As shown in Fig. 4, the Collabrary uses a markedly different architecture for transmitting multimedia. The centerpiece of this architecture is the *shared dictionary*. This distributed data structure blends programming idioms from notification servers [10], groupware programming [13], distributed shared memory systems [21], Model-View-Controller architectures [4], and filesystems. In the remainder of this section, we illustrate how this shared dictionary is used to rapidly prototype multimedia groupware.

```
/* Initialisation */
Hashtable windList=new …;
Camera camera=new …;
camera.Size=…320x240;
camera.FrameRate = 10;
VideoCodec codec=new …;
codec.Open("MJPEG",320,240,…);
SharedDictionary sd=new …;
sd.Open("tcp://www.host.com:video");

void sd_Opened(…) {
  /* Tie data to connection status */
  sd[sd.Me+"/.transient"]=sd.Me;
  /* Store a user display name */
  sd[sd.Me+"/name"]="Mike";
}

void sd_Closed(…) {
  /* Prompt user to reconnect */
  if(sd.Troubleshoot(…)) {
    retries=1;
}}

void sd_Entered(string id) {
  /* Create separate GUI window */
  VideoWin win=new VideoWin(id);
  win.Show();
  winList[id]=win;
}

void sd_Exited(string id) {
  /* Dispose of GUI window */
  VideoWin win=windList[id];
  win.Close();
  windList[id]=null;}

void camera_Captured(IPhoto curFrame) {
  /* Store compressed video frame */
  sd["/user/"+sd.Me+"/video"]=
    codec.Compress(curFrame);
}

public VideoWin(string id) {
  /* Set window caption */
  this.Text=sd[id+"/name"]…;
  /* Subscribe to video */
  Subscription video=sd.Subscribe(id+"/video");
  video.Notified+=…video_Notified…;
}

void video_Notified(…object val…) {
  Photo p=videoCodec.Decompress(val…)…;
  pictureBox.Image=…p…;
}
```

Fig. 4. Implementing an *n*-way video conferencing application with the Collabrary shared dictionary.

A. *Centralized Server Network Architecture*

End-programmers do not need to think about the setup of the shared dictionary network, as the `SharedDictionary` object they use to access it takes care of all the details. This keeps end-programmers focused on the structure of the data they wish to share, not the mechanics of sharing it.

Internally, this object uses a client/server architecture for a centrally-coordinated data store. Clients send updates to the server, which orders them and forwards them to other clients. Data is cached at each client for rapid access.

To the end-programmer however this object looks like a hash table that maps hierarchically structured keys—text strings resembling paths in a conventional disk file system—to values. The object manages the connection to the server transparently, automatically marshalling data sent.

The shared dictionary automatically deals with *late-comers* by providing a client with a completely up-to-date version of the data store at the time it connects to the server, similar to [11]. The `Opened` event is raised on the client after it has connected and fully updated its local cache. In the figure, the handler for this event stores a "display name" for the current client which is used as a window caption on other clients.

When the connection is closed or broken due to a network connectivity problem, the end-programmer can handle the `Closed` event and set a flag to have the connection automatically re-established. The code in the figure uses the `Troubleshoot` method to notify the end-user of connection troubles and ask for permission to reconnect.

When a client connects to the shared dictionary server, the server informs the other clients already connected to it, and they in turn each raise the `Entered` event. In this simple example, a separate window is created to display the video from each client. This window will be deleted in the `Exited` event handler when the corresponding client disconnects.

B. *Organizing & Storing Data in a Hierarchical Dictionary*

Values that may be stored in the shared dictionary may be of practically any type. The Collabrary automatically marshals the data i.e., convert it into byte array that can be transmitted over a network. This makes the shared dictionary:

- *accessible*, because novice programmers need not concern themselves with marshalling;
- *lightweight*, because expert programmers need not write any code to take care of marshalling; and,
- *flexible*, because data are shared in their normal types.

A value is stored using a simple assignment syntax e.g., `sd["/user/name"]="Mike"`. The value is removed by overwriting it with `null`. This is:

- *accessible*, because it is the same syntax as that which is used with the system-supplied hash table class;
- *lightweight*, because assignment is one of the simplest programming statements; and,
- *flexible*, because the end-programmer decides the names of keys and the values stored at each.

The shared dictionary supports hierarchical organization of data because keys look like paths in a disk filesystem. In the figure, the `SharedDictionary.Me` property retrieves the current connection's id and prefixes it to the `"/video"` substring to generate the complete key used to transmit compressed video frames.

C. *Subscription Notifications & the MVC Architecture*

The Collabrary shared dictionary has a mechanism whereby the end-programmer can request notification of changes made to the dictionary. The end-programmer obtains a `Subscription` object, specifying a key or pattern of keys to watch, and handles the `Notified` event on it. The simple pattern matching language available resembles the "filename globbing" pattern matching language used in UNIX and related disk file systems. (The code in the figure does not need to make use of pattern-based subscriptions.)

Video is streamed by repeatedly storing individual video frames at the same key in the shared dictionary. The server broadcasts the updates to all connected clients. As each update is received, the key is inspected and the `Notified` event handler for any matching subscription is invoked with parameters that describe the change. In the figure, a separate subscription is used to decompress the compressed video from each client and render it into its own GUI window.

The ability to organize data hierarchically and receive asynchronous notification of data changes allows the end-programmer to employ the shared dictionary as the "model" within a Model-View-Controller or Presentation-Abstract-Control architecture pattern [4]. These models are important because they allow the end-programmer to separate the abstract data model from how it is gathered (i.e., the input gathered by the controller) and how that data is displayed (via the view or presentation). This separation is critical in a distributed environment where different clients may have different views or different means of managing user input.

D. Controlling Presence Distribution of Keys & Values

The Collabrary shared dictionary includes features to control how long keys or values stay in the shared dictionary. Normally, when a client puts a value in the shared dictionary, it is sent to all clients and it is stored in the dictionary indefinitely. It can be overwritten (by any client, not just the one that first put it there) by assigning a new value to the same key. The entry is removed only when a client sets it's value to `null`. A client receives a copy of all data on the server and does not need to obtain a subscription for it or otherwise express interest in it. However, the shared dictionary server may silently drop an unsent and unneeded update when the link to a particular client is slow or congested.

The default persistence and distribution behavior is good for most purposes, but may be changed to make the prototype more robust in lower-bandwidth network conditions. Several options are available to:

- control data caching;
- receive only updates for keys it has subscribed to;
- ensure every update (even redundant ones) are received;
- send high priority data preemptively;
- specify which other clients receive the data;
- indicate how long data stays in the cache; and,
- tie the presence of keys in the cache to the connection status of a particular client.

For example, Fig. 4's `Opened` event handler stores a flag in the dictionary that binds persistence of the subtree used to store a client's data to the connected status of the client. The server removes the subtree when the client disconnects.

VII. Discussion And Conclusions

We believe that the Collabrary is a significant contribution to rapidly prototyping multimedia groupware because it *trivializes* four common programming tasks for multimedia groupware: capturing, manipulating, transmitting and rendering multimedia. This is illustrated in three ways. First, we illustrated how the Collabrary is *accessible* because it allows end-programmers to use programming idioms that are already familiar to them. Second, we have shown how the Collabrary is *lightweight* and makes "simple things simple" in a several ways. Third, we explained how the Collabrary is *flexible*, where it makes "complex things possible".

While space does not allow us to elaborate, the above design features have been validated in practice. The Collabrary has seen active use for several years by a variety of researchers. It is the architecture underneath several long-running and heavily used media space prototypes, e.g., the Notification Collage [1], Community Bar [22], and Home Media Space [23]. It is the basis of several quite novel systems, such as *mixed presence groupware* [24] and user interfaces for generating custom notifications [25]. It was used to teach undergraduates groupware programming, where students designed and quickly implemented many intriguing systems [6] in a very short amount of time.

However, we recognize that some will see the Collabrary as just another toolkit. Perhaps the more long-lasting contribution is our design requirements: we believe any universally accepted prototyping toolkit for distributed multimedia groupware research must trivialize four common programming tasks—capturing, manipulating, transmitting and rendering—by being accessible, lightweight, and flexible. The Collabrary merely shows one way that this can be accomplished.

Try it yourself. The Collabrary may be downloaded from http://grouplab.cpsc.ucalgary.ca/collabrary.

References

[1] S. Greenberg & M. Rounding, "The Notification Collage: Posting information to public and personal displays", in *Proc. ACM CHI*, 515-521, 2001.

[2] A. Fass, J. Forlizzi, & R. Pausch, "MessyDesk and MessyBoard: Two designs inspired by the goal of improving human memory" in *Proc. DIS*, 303-311, 2002.

[3] G.Fitzpatrick, *The Locales Framework: Understanding and Designing for Wicked Problems,* Kluwer Academic Publishers, 2003.

[4] *Human Computer Interaction, Second Edition,* Dix, Alan, Finlay, Abowd, & Beale eds., Prentice Hall International, 1998.

[5] W.A.S. Buxton "Living in augmented reality: Ubiquitous media and reactive environments", in *Video Mediated Communication*, Finn, Sellen, & Wilbur eds., Lawrence Erlbaum Associates, 363-384, 1997.

[6] S. Greenberg, "Toolkits and interface creativity", *Multimedia Tools and Applications*, Kluwer Academic Publishers, in press.

[7] *RTP: A Transport Protocol for Real-Time Applications*, IETF RFC 3550, 2003.

[8] *SIP: Session Initiation Protocol*, IETF RFC 3261, 2002.

[9] *Web Services Architecture*, W3C Working Group Note 11, Feb. 2004.

[10] G. Fitzpatrick, S. Kaplan, T. Mansfield, A. David, & B. Segall "Supporting public availability and accessibility with Elvin: Experiences and reflections", in *J CSCW*, 11:3, 447-474, 2002.

[11] R. Burridge, *Shared Data Toolkit for Java Technology User Guide*, Sun Microsystems JavaSoft, 2004.

[12] S. McCanne, E. Brewer, et. al, "Toward a common infrastructure for multimedia-Networking Middleware, in *Proc. NOSSDAV'97*, 1997.

[13] N. Roussel, "Exploring new uses of video with videoSpace", in *Proc. EHCI*, LNCS 2254, 73-90, Springer, 2001.

[14] M. Roseman & S. Greenberg, "Building real time groupware with GroupKit", in *ACM TOCHI*, 3:1, 66-106, 1996.

[15] *Microsoft NetMeeting, 3.01 SDK*, Microsoft Corporation, 2003.

[16] *Microsoft DirectShow 9.0 SDK*, Microsoft Corporation, 2005.

[17] *Java™ Media Framework API Guide*, Sun Microsystems, 1999.

[18] G.R. Bradski, "Computer video face tracking for use in a perceptual user interface", in *Intel Technology Journal Q2'98*, 1998.

[19] C. Gutwin & R. Penner, "Improving interpretation of remote gestures with telepointer traces", in *Proc. ACM CSCW 2002*, 49-57, 2002.

[20] *Video coding for low bit rate communication*, ITU-T H.263, 2005.

[21] N. Carriero & D. Gelernter, "Linda in context", in *Comm. ACM*, 32:4, 444-458, 1989.

[22] G. McEwan & S. Greenberg, "Community Bar: Designing for awareness and interaction", in *ACM CHI 2005 Workshop on Awareness systems: Known Results, Theory, Concepts and Future Challenges*. 2005.

[23] C. Neustaedter & S. Greenberg, "The design of a context-aware home media space", in *Proc. UBICOMP*, Springer-Verlag, 297-314, 2003.

[24] A. Tang, M. Boyle, & S. Greenberg, "Display and presence disparity in mixed presence groupware", in *JRPIT*, 37:2, 71-88, May 2005.

[25] S. Greenberg, & M. Boyle, "Tracking visual differences for generation and playback of user-customized notifications" Report 2005-777-08, Dept. of Computer Science, University of Calgary, Canada, April, 2005.

Modelling complex user experiences in distributed interaction environments

Fabio Pittarello and Daniela Fogli

Abstract— **The focus of this work is that of the so-called mixed reality domain, where interaction is increasingly becoming a complex matter in which the user navigates the different locations of a 3D virtual or real world, manipulates objects and accesses content with different degrees of heterogeneity and synchronization. In order to control such complexity we specify the 3D environment by a multilevel control finite state machine using the statecharts notation and we model the interaction with the environment and the multimedia content communicated to the user through the concepts of *experience process* and *experience pattern.* A set of distributed software agents log and process data related to the user experience for controlling interaction and data presentation at run-time.**

Index Terms— **3D environment, software agents, user experience, experience pattern, interaction pattern.**

I. Introduction

Interactions in 3D environments is increasingly becoming a complex matter where users navigate the scene in order to complete different goals. This is true for all the segments of the so-called mixed reality [13] domain, characterized by different mixtures of real and virtual 3D elements that the user navigates with her/his real body or virtual counterpart (i.e., avatar). Given the complexity of such environments, the term *experience*, usually referred to human activity in real life, can be successfully applied. In fact, a variety of features, such as subjective involvement in the scene and progressive evolution of the user behavior on the basis of the activity previously done still characterizes the human behaviour in mixed reality environments. Such environments are typically characterized by interactive objects distributed into the scene; the user experience itself is the result of an interaction distributed across the different locations of the environment. Monitoring and controlling satisfactorily such activity with a set of sensors and actuators distributed over the environment is a complex task that may be simplified by an accurate modeling of the user experience. Using such model may result in augmenting overall user satisfaction and preventing interaction errors.

D. Fogli is with the Dipartimento di Elettronica per l'Automazione at the University of Brescia, Brescia Italy (e-mail: fogli@ing.unibs.it).

F. Pittarello is with the Dipartimento di Informatica at the University of Ca' Foscari, Venice Italy (pitt@unive.it).

Our approach capitalizes on the Interaction Locus (IL) model [15][16] and on the concepts of *interaction process* and *interaction pattern,* previously introduced by the authors; besides, the approach goes a step beyond by discussing the concepts of *experience process* and *experience pattern* that permit to extend the typologies of experiences that can be modeled. An *interaction process* [4] is a detailed logging of user activity and system reactions. *Interaction patterns* are recurrent subsequences of an interaction process; their recognition is useful both for controlling and proactively adapting user interaction. While interaction patterns allow to face with complex situations, the approach fails where an explicit control of content fruition is required. An increasing number of experiences in 3D worlds (e.g., educational experiences) reserve an important part to content fruition. In such experiences the explicit knowledge of the part of content browsed by users (e.g., text, images, sounds and also parts of the 3D environment itself) becomes an important input for the system that can affect the evolution of the experience itself.

The need of having a more sophisticated means to model the overall user activity, including a detailed view of multimedia content browsed, led us to define the concepts of *experience process* and *experience pattern*. Such concepts were introduced in [17]; the present work provides a formal specification of the concepts, and a detailed presentation of the multilevel control finite state machine (CFSM) specifying 3D environments. A description of the relations between such machine and the logging processes that permit to control and simplify the user experience is also presented.

Our approach takes advantage of a set of coordinated agents distributed into the environment for monitoring and controlling the user experience. Such approach is useful also for optimizing system performance and is suitable to mixed reality paradigms characterized by distributed intelligence.

The paper is organized as follows: Section II describes related work. In Section III we summarize previous results for the definition of structured 3D environments monitored by a set of coordinated agents; the concepts of interaction process and pattern are presented. In Section IV we introduce the concepts of experience process and pattern. Section V describes three case studies; a more formal description of the environment and of the user experience for one of such cases is given in Section VI. Some final remarks conclude the paper.

II. Related works

A key component of our proposal is the introduction of a set of *agents* [11], which act in the virtual/real world keeping track of the initial background of the human involved in the experience, of the relevant interaction and of the kinds of multimedia content enjoyed by the users during their navigation within the 3D environment. Agents have been considered in the context of intelligent workspaces for organizing the use of different devices; application examples include systems that enhance real environments for managing meetings and retrieving information through a multi-modal interface [7].

Structuring 3D environments is an important step for controlling them. Some approaches [15] establish at design time the relations between the morphology of space and activity performed, other approaches try to derive semi-automatically the partitions of space (i.e., *activity zones* [10]) from the observation of user actions. In any case, defining zones for different classes of activity can be useful for defining tasks and areas of influence of a set of agents distributed into the environment.

Navigational modalities have been explored in the hypermedia realm [5] leading to define and underline significant paradigms such as indexes and guided tours for moving through content. Modeling navigation inside 3D scenes is more complicated, due to the augmented degrees of freedom potentially offered by such environments. A wide range of paradigms ranging from free navigation to different types of constrained navigation (i.e., guided tours or other paradigms where the orientation and/or the translation of the user viewpoint are partially or totally controlled by the system) have been defined. While free navigation paradigms are useful to explore the environment in detail, recent research shows that constrained navigation can be helpful for supporting user to complete their task and for avoiding them to get lost [6].

Not only structuring 3D environments, but also formally specifying the interaction of the users with them is crucial to obtain a better control. Several research groups have proposed systemic models of the Human-Computer Interaction (HCI) processes. Barnard et al. describe the user-computer interaction process as a *syndetic* system, i.e. a system binding up subsystems of different nature, the human as a cognitive subsystem, on the one hand, and the computer as a computing artefact, on the other [1]. Our work is based on the model proposed in the past by the Pictorial Computing Laboratory (PCL) [2] and successively extended in [3]. It regards HCI as a process in which the user and the computer communicate by materializing and interpreting a sequence of messages at successive instants of time. The activities performed by the computing sub-system are formalized by taking into account this symmetry and that two interpretations of each message exist, one by the human and one by the computer.

III. Background

This section briefly introduces models and concepts on which this work is based. In particular, it capitalizes on the Interaction Locus (IL) model [15][16]. The IL approach aims at giving a structure to 3D environments belonging to all the segments of the mixed reality domain. The approach, originally meant to support user navigation, has been progressively developed to support authoring, user profiles, access to information with different devices and user interaction. Basically the 3D environment is divided into a set of locations characterized by a recognizable morphology; a set of allowed/forbidden user interactions and multimedia content are associated to each IL. The IL has gradually changed its nature to include not only information necessary for navigation (e.g., labels and sounds identifying locations), but also complex content useful for augmenting the user's knowledge. ILs are organized in hierarchies and the properties of a given level can be inherited by the inner one.

The observation of the user interaction within a set of ILs is at the basis of the approach proposed here, aimed at controlling and adapting both the interaction and the fruition of multimedia content. In previous works we defined the concept of *interaction process* and *interaction pattern* [4]. The formalization of the concept of interaction pattern was obtained by adopting the PCL model of HCI [2]. Some details on the specification of interaction within ILs are here introduced for the sake of explanation.

In the mixed reality environments humans and computer systems communicate exchanging multimodal messages; computer messages may be the images which appear on the screen display of a computer or a palmtop, the sound from speakers, the mixed scene composed by virtual and real elements with which the user may interact. A human interprets the image on the screen or the sound from audio speakers or the force from a haptic device by recognizing characteristic structures. More precisely:

Def. 1: A *characteristic structure* (cs) is a set of system generated events which may be perceived by the user as a functional or perceptual unit.

The css are materialized on the output devices to become perceptible by the users; they are the output events of the virtual entities [3]. Virtual entities are used by the designer to specify the system structure and behavior:

Def. 2: A *virtual entity* (ve) is a virtual dynamic open system that exists only as the result of the execution of a program P by a computer.

Actually, P is a system of programs, some of which (*Input programs*) acquire the input events generated by the user actions, some compute the ve reactions to these events (*Application programs*), and some output the results of this computation (*Output programs*). More precisely:

Def. 3: A *program* is specified as $\mathsf{P} = \langle \mathsf{In}, \mathsf{Ap}, \mathsf{Out} \rangle$, where In denotes the input programs, Ap denotes the application programs, Out denotes the output programs.

During an interaction, the user operates on some input device to manifest his/her requirements or commands to the ve. The ve captures input events generated by user actions and reacts to them generating output events toward users (i.e., the css).

More precisely a ve is defined on input and output alphabets.

Def. 4: The *input alphabet* A of a ve is a finite set of user activities.

Def. 5: The *output alphabet* O of a ve is the set of possible css generated by the ve as output events.

The user activity is defined as follows:

Def. 6: A *user activity* is specified as $\mathsf{a} = \langle \mathsf{op}, \mathsf{cs} \rangle$, where op - operation - denotes the sequence of events perceived by the machine as a consequence of the user action on some input device at a given step of the interaction, and cs is a char-

characteristic structure of a ve.

Given a ve, its current state is called characteristic pattern.

Def. 7: A *characteristic pattern* (cp) is specified as cp=<cs, d, <int, mat>>, where the characteristic structure (cs) is a set of user perceivable events managed by the In and Out programs, d is a suitable description of the state of the programs Ap, int (interpretation) is a function, mapping the current cs onto d and mat (materialization) is a function mapping d onto cs.

A ve may thus be specified as a dynamic open system.

Def. 8: A virtual entity is a 5-tuple ve = <S, O, f, η, s_0> on the input alphabet A, where

1. S is the set of admissible ve states, i.e. the set of its cps;
2. O is the output alphabet of the ve;
3. f: $A \times S \rightarrow S$ is the next state function;
4. η: $A \times S \rightarrow O$ is the output function;
5. $s_0 = cp_0$ is the initial state of the ve.

This kind of specification can be described in a diagrammatic way through the use of a control finite state machine (CFSM). As a consequence, the following definition of interaction process can be provided.

Def. 9: An *interaction process* is a sequence of triples <s_t, a_t, s_{t+1}>, where s_t is the state of the CFSM, a_t is the activity performed by the user at a certain time t, and s_{t+1} is the new state of the ve after the user activity is captured and managed.

From this definition, that of interaction pattern can be derived.

Def. 10: An *interaction pattern* is a recurring sub-sequence of an interaction process.

A software architecture based on agents [4] have been proposed to log the interaction process, recognize the recurring interaction patterns and proactively adapt interaction with ILs, trying to anticipate the user needs. In particular, two kinds of agents have been defined, one associated to each IL, and the other associated to the user. The first agent, called *genius loci*, takes care of the place by giving the visitors the opportunity to get most benefit from its exploration. The latter, called *numen*, follows the user during navigation by accumulating and managing knowledge about him/her. The numen knows the user profile, accumulates the exploration history across several places, and is able to interact with the genii of the different places in order to give them information about how to help the user in his/her visit. The two agents mediate the interaction between the user and the environment accumulating, maintaining and exchanging knowledge about the user and the interaction place.

IV. Introducing the Experience Pattern

The need for a more detailed monitoring of the user activity inside the 3D environment rises from the introduction of complex experiences where content fruition is a substantial part of the experience itself and may affect the rest of the interactive session. In such experiences the distributed and cooperating agents need a means for sharing the knowledge of the content browsed, in order to control the experience. The concepts of *experience process* and *experience pattern* satisfy such requirements. In fact, while the interaction process represented the path of the user among the different interaction opportunities and recorded simply the interaction steps (e.g., the user clicks button1 inside IL1), the experience process represents the path of the user among the different interaction and content fruition opportunities and records both interaction steps and content browsed (i.e., at least a summarization of it). Recurring sequences of an experience process can then be extracted by the agents giving indications about users' habits, preferences and information needs. We call such sequences experience patterns. An additional task for agents, introduced in this work, is the progressive building of a map of the user's knowledge. The result of such activity can then be used for enabling appropriate control or for proactive behaviour; in both cases the target of agents' action can be content presentation or user interaction. On the basis of these premises, we may give the following formal definitions of experience process and experience pattern:

Def. 11: An *experience process* is a sequence of 4-tuples <s_t, a_t, s_{t+1}, $content_{t+1}$>, where s_t is a state of the 3D environment, a_t is the input activity performed by the user, s_{t+1} is the state reached by the 3D environment as a reaction to a_t, $content_{t+1}$ is a (machine-readable) description of multimedia content enjoyed by the user as a consequence of his/her activity a_t.

Def. 12: An *experience pattern* is a recurrent sub-sequence of the experience process.

Therefore, experience patterns are still sequences of 4-tuples <s_t, a_t, s_{t+1}, $content_{t+1}$> that can be extracted from the experience process for example by exploiting literature algorithms such as those described in [12]. Interaction patterns, can be described alternatively as recurrent sub-sequences of the experience process projected on the state and activity dimensions, becoming therefore sequences of triples <s_t, a_t, s_{t+1}> coherently with Definitions 9 and 10.

V. Three Case Studies

In order to make more clear the concepts described above, the following sections will present three case studies where such concepts can be successfully applied; a more formal description related to the second case study, evidencing the relations with the multilevel CFSM specifying the 3D environment, will follow in Section VI. The three case studies are classified according to the complexity of the navigation models available for the different kinds of experience proposed. Figure 1 illustrates the meaning of symbols used in figures associated to the examples.

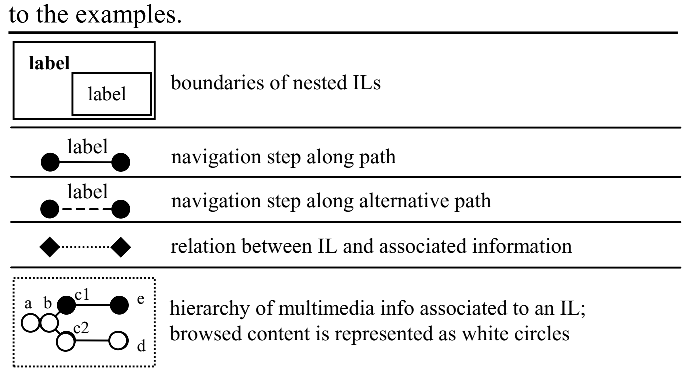


Figure 1. Explanatory list of the symbols.

Guided tour. Such navigation modality, where the user follows the steps of a predefined path, reflects the conceptual order conceived by the author of the experience for browsing the 3D world and accessing content; it is meant for all the categories of users, with a particular reference to novices that need assistance for moving across a complex environment. Figure 2 shows the layout of the environment (i.e., a cultural exhibition area organized in rooms containing works of art) and the set of hierarchical ordered ILs with associated multimedia information. In general, multimedia information has a hierarchical structure, as shown in Figure 1; hypertextual paths among the different informational nodes may be more complex, but they are not shown for the sake of simplicity. The user experience is the result of his/her navigation in the 3D world and of his/her content browsing choices inside the multimedia information hierarchies. The navigation steps done by the user inside the environment are ordered progressively from s1 to s10. In the example, at first user enters the exhibition area 1 (a first level IL) and browses information chunks a, b and c. Then s/he enters the room 1.1 (a second level nested IL) and browses a part of the associated content (e.g., a, b, c2, d1 and d2); the path then leads him/her to the object 1.1.1 of the room (i.e. a third level nested IL) giving him/her the opportunity to browse its content. The visit of the environment prosecutes till the end according to the design of the authors. Note that the browsed content is only a portion of the available multimedia information. In this situation the set of agents log the user activity into the experience process and use it to adapt the content presented to the user both basing their action on classification of information or on text indexing.

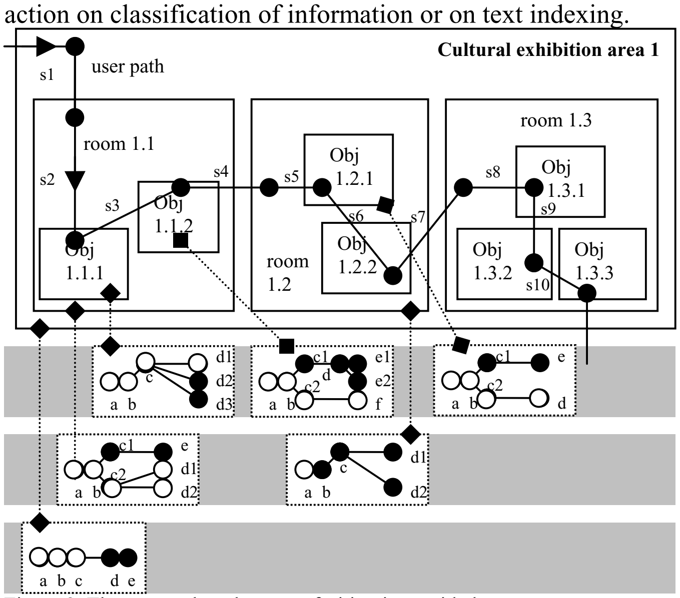


Figure 2. The user path and content fruition in a guided tour.

In the first case a preliminary organization of information nodes in classes is required. The coordinated agents will recognize the recurring experience patterns and will present to the user information proactively selecting the classes of information s/he has demonstrated to be more interested in. For example, if the user has demonstrated constant interest for a specific class of information (e.g., artist's biographical sketch) browsed in node d1 of object 1.1.1 and in node f of object 1.1.2, when s/he will approach object 1.2.1 s/he will be presented with information node d belonging to the same class. In the latter case content adaptation is based on the indexing of text contained in multimedia nodes, performed by genii loci as a part of their monitoring work. Indexed information is saved into the experience process and used by the numen for building a progressive map of the user's knowledge. Such map is then shared with the genii of the following ILs that will compare it with indexed information related to the locations controlled by them; at the end, they may proactively select and present information on the basis of content matching. In both cases, according to the well-known usability guidelines [14], proactive behaviour should be coupled with the option to go back to a default entry point (e.g., the root of the multimedia information hierarchy for the current IL) that the user may select in case s/he gets disoriented or lost.

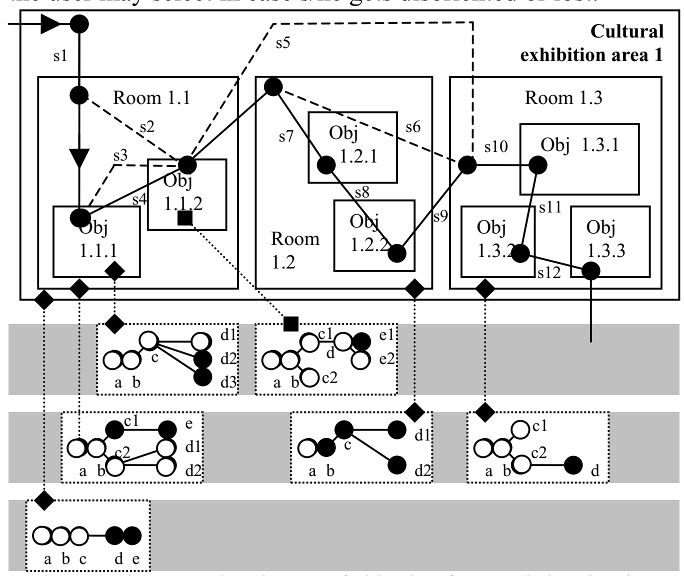


Figure 3. The user path and content fruition in a free wandering situation.

Free wandering. Such navigational paradigm is meant for expert users that decide their own path along the 3D world; such modality enhances the agents' activity, for selecting different content in relation to previous user movements. Figure 3 shows an example of 3D environment where the user is not constrained to a specific navigation path. In the example the user decides in some occasions to deviate (dashed line) from the suggested path (black line). The navigational steps done by the user are shown in the figure, ordered progressively from s1 to s12. In such experience, genii loci log navigation done and content browsed into the experience process and communicate such information to the numen; the numen progressively builds a map of the knowledge acquired by the user. Such knowledge is shared with the genius of the following IL, that may suggest the user, after having compared the user knowledge with the requirements for that IL, to come back to acquire missing information. Alternatively, the genius loci may proactively shorten the path along the associated information hierarchy if the user knowledge already incorporates notions that are also redundantly available for that IL.

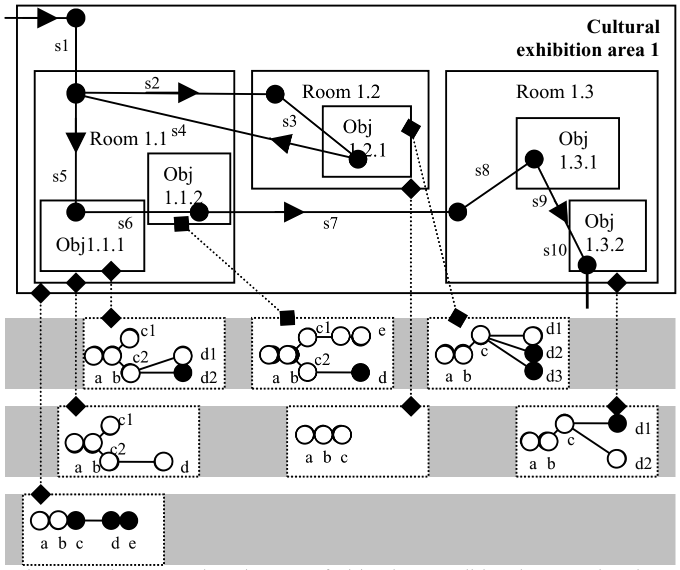


Figure 4. The user path and content fruition in a conditional access situation.

Conditional access. In this case access to different zones is conditioned to the satisfaction of certain requirements related to user activity in other ILs of the environment. This is a typical situation in e-learning environments where the user has to prove to be able to perform certain operations or to have acquired a knowledge about specific arguments.

Figure 4 shows an example of 3D learning environment where the user moves through the different zones of the environment augmenting his/her knowledge and solving questions in order to proceed. For example the user entering room 1.1 is invited to go to room 1.2 for answering to a set of questions and gaining full access to room 1.1 content. Navigation through nodes of associated multimedia information is regulated by genii on the basis of previous behaviour (e.g., the default path through information nodes for user entering room 1.1 is a-b-c1; such path changes to a-b-c2-d1/d2 when the user comes back after having answered correctly to questions contained in room 1.2). Generally speaking, each genius loci detains the control of a specific zone and the knowledge of the requirements for allowing access to users. Again the numen progressively builds, on the basis of the experience process monitored by the genii of the visited ILs, a map of the interactions done by the user and of the knowledge acquired by him/her. The result of its activity is shared with the current genius loci for allowing it to act appropriately, allowing/denying access and presenting appropriate information. This scheme enables the creation of multi authored e-learning experiences, where each author detains the knowledge related to a certain zone of the learning path and establishes the requirements for accessing it.

VI. Formal Description of the User Experience

Following the definitions given in Sections III and IV, we will give a more formal description of the environment and of the user experience described for the free wandering case. We characterize each IL as a virtual entity. Each ve, representing an IL of a given level, may be composed by other ves representing nested ILs of the lower level; such hierarchical structure can be easily specified using statecharts [8] that permit to model systems at different levels of abstraction. This specification allows us to provide a formal definition of the environment and of the experience processes. In the case at hand, ILs in the 3D environment belong to three possible levels: 1) cultural exhibition area; 2) room; 3) interaction object. Each kind of IL provides some interaction possibilities allowing the user to change the state of the environment and to enjoy the multimedia content associated with the IL.

The statechart describing a 3D environment with three cultural exhibition areas can be represented at a high level of abstraction as in Figure 5, where the specification does not give any details about the rooms composition. We adopt here the notation proposed in [9]. Note the history symbol H, which is exploited to remember the last visited state belonging to the lower level; such information permits to take it as the initial one after the first visit.

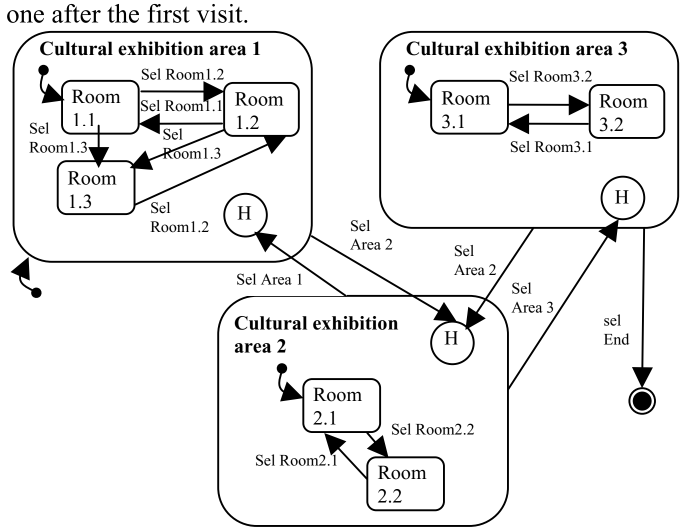


Figure 5. The statechart specifying the 3D scene at a high level of abstraction.

Actually, each lower level state of an exhibition area represents a set of states of a virtual entity corresponding to an IL of type "room". Such IL can be described, at a lower level, by the interaction objects behaving in the room. Figure 6 illustrates the lower level statechart specifying a room containing two independent interaction objects represented as two state diagrams that run concurrently.

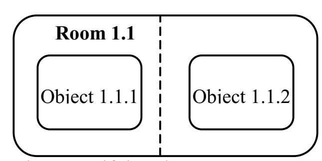


Figure 6. The statechart specifying the concurrent components of a room.

Each interaction object is in turn an IL and thus can be specified as a virtual entity described by a further lower level state diagram. An example is shown in Figure 7 that illustrates the statechart specifying an interactive object (e.g. a painting) with which it is possible to interact to see its description or the author's biography.

Moreover, the description may include hyperlinks associated to some keywords: the interaction with them leads the object to move in a new state, by providing the user with new multimedia information.

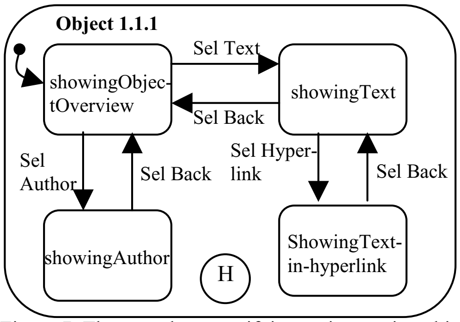


Figure 7. The statechart specifying an interactive object.

As explained in the previous section, at each level of the 3D environment, the user may enjoy different kinds of multimedia content. Multimedia content is obtained as a consequence of the user performing an activity, i.e. an operation with respect to a characteristic structure. As previously mentioned, each user activity is captured by the ve to which the cs belongs, and, besides determining a possible state change, it also fires the execution of a computational function providing the associated multimedia content. Therefore, in terms of the statechart, the multimedia content is the result of the output function. Each state s of the statechart is described by an array [a, b, c, ..., z] of variables, each one associated with a different nested level of the statechart. For example, in the example described above, s can assume the value [cultural exhibition area 1, room 1.1, object 1.1.1, showingAuthor], and in this case the associated content is an indexed version of the multimedia content describing the author biography, that is a synthetic description of the multimedia content provided to the user. As already mentioned, multimedia content can be associated with any kind of IL. For example, when the user is in the cultural exhibition area 1 and enters room 1.2, it might enjoy the multimedia content associated with the room, e.g. a particular music related with the theme treated in the room. In this case, state s_t is [cultural exhibition area 1, room 1.1, null, null], a_t is the activity of entering the room 1.2, s_{t+1} is [cultural exhibition area 1, room 1.2, null, null], and $content_{t+1}$ is an indexed version of the music title. (The symbol null means that the variables indicating the interaction objects and their states have not a meaningful value). Concerning multimedia elements such as music, we require that each element associated with ILs would have an alternative textual form expressing the same content (or at least a summarization of it); this requirement is coherent with accessibility guidelines [18] for web hypermedia and grants an easy conversion and/or indexing of heterogeneous elements. The knowledge of the statecharts is distributed among the genii loci. Such decentralized knowledge is used cooperatively by the agents to log the experience process, extract the experience patterns and build progressively the map of the user knowledge. The final goal of this process is to control and proactively adapt both navigation across distributed locations and fruition of multimedia content embedded in the interaction objects distributed in the environment.

VII. CONCLUSION

The approach presented in this paper represents an improvement of the previous work based on the concepts of experience process and experience pattern, permitting to describe and control a wider range of experiences in 3D environments, including those ones where content fruition is an important part of the user activity and can determine its evolution. A component based architecture based on the concepts discussed in this work is described in [17]. A pilot study on a scaled-down implementation of such architecture is currently being performed. The results of this study, focusing both on quantitative parameters (e.g. how long users take to perform a task with proactive features enabled) and qualitative factors (e.g. user satisfaction) encourage us to design an experimental system to verify all the features described in this work.

VIII. REFERENCES

[1] Barnard, P., May, J., Duke, D., Duce, D., Systems, Interactions, and Macrotheory. *ACM Transactions on HCI*, 7(2), 2000, 222-262.

[2] Bottoni, P., Costabile, M. F., Mussio, P., Specification and Dialog Control of Visual Interaction. *ACM Transactions on Programming Languages and Systems,* 21(6), 1999, 1077-1136.

[3] Costabile, M.F., Fogli, D., Fresta, G., Mussio, P., Piccinno, A., Software Environments for End-User Development and Tailoring, *Psychology*, 2(1), 2004, 99-122.

[4] Fogli, D., Pittarello, F., Celentano, A. Mussio, P., Context-aware interaction in a mobile environment. *Proc. Fifth International Symposium on Human Computer Interaction with Mobile Devices and Services (MobileHCI03)*, Udine, Italy, September 2003, 434-439.

[5] Garzotto, F., Paolini, P., Schwabe, D. – HDM – A Model-Based approach to hypertext application design, *ACM Transactions on Information Systems*, Vol. 11, No. 1, Jan 1993, pp. 1-26.

[6] Haik, E., Barker, T., Sapsford, J., Trainis, S., Investigation into Effective Navigation in Desktop Virtual Interfaces. *Proc. 7th International Conference on 3D Web Technology*, Tempe, Arizona, USA, 2002, 59-66.

[7] Hanssens, N., Kulkarni, A., Tuchida, R., Horton, T., Building Agent-Based Intelligent Workspaces. *Proc. International Conference on Internet Computing,* 2002, 675-681.

[8] Harel, D., Statecharts: A Visual Formalism for Complex Systems. *Sci. Computer Prog.*, July 1987, pp. 231-274.

[9] Horrocks, I., *Constructing the User Interface with Statecharts*, Addison-Wesley, 1999.

[10] Koile, K., Tollmar, K., Demirdjian, D., Shrobe, H. E., Darrell, T., Activity Zones for Context-Aware Computing. *Proc. Ubicomp* 2003, 90-106.

[11] Jennings, N. R.. An Agent Based Approach for Building Complex Software Systems. *Communications of the ACM*, 44(4), 2001, 35-41.

[12] Lelewer, D. A., Hirschberg, D. S., Data Compression. *ACM Computing Surveys*, 19(3), 1987, 261-296.

[13] Milgram, P., Kishino, F., A Taxonomy of Mixed Reality Visual Displays, *IEICE Transactions on Information Systems*, Vol. E77-D No. 12, 1994, 1321-1329.

[14] Nielsen, J., *Usability Engineering*, Academic Press, 1993.

[15] Pittarello F., Multi sensory 3D tours for cultural heritage: the Palazzo Grassi experience, *Proc. ICHIM2001, Cultural heritage and technologies in the 3rd millennium*, Milano, Italy, 2001, 73-90.

[16] Pittarello, F., Accessing Information Through Multimodal 3D Environments: Towards Universal Access. *Universal Access in the Information Society Journal*, 2(2), 2003, 189-204.

[17] Pittarello, F., Context-based Management of Multimedia Documents in 3D Navigational Environments, *Proc. MIS 2005, Workshop on Multimedia Information Systems*, Sorrento, Italy, September 2005 (in press).

[18] Web Accessibility Initiative, http://www.w3.org/WAI/

A Robotic Interface for Retrieval of Distributed Multimedia Content

Ellen Lau
Interactions Lab
Dept. of Computer Science
University of Calgary, Canada
laue@cpsc.ucalgary.ca

Ehud Sharlin
Interactions Lab
Dept. of Computer Science
University of Calgary, Canada
ehud@cpsc.ucalgary.ca

Ariel Shamir
The Interdisciplinary Center
Efi Arazi School of Computer Science, Israel
arik@idc.ac.il

Abstract

In a social setting, it is often beneficial to have a partner that is alert of the environment and the circumstances and is able to communicate information that otherwise does not come to our awareness. Typically, these are human companions, however a robotic partner may be an alternate solution. For a robotic partner to be successful, it must naturally interact with its user, and be able to retrieve data and analyze its content. In this paper we present Iolaus, a robotic partner for social settings. Iolaus can be implemented as a robotic parrot perched on the user's shoulder, or as an attentive companion similar to a Seeing Eye dog. When the user enters a new social setting the robot can be asked and should attempt to answer questions regarding people and surroundings, very similar to a human partner. In this phase of the project, we are using the Sony Aibo robot dog to implement Iolaus's information retrieval component demonstrating how a human-robot interface can help the user access a vast amount of distributed information and multimedia content in a socially acceptable manner.

Keywords: human-robot interaction, distributed information retrieval, multimedia human computer interaction.

1. Introduction

The growth of mobile computing devices sees with it a growth in distributed computing. Gone are the days when mobile appliances were limited to performing simple tasks. Today, mobile devices are capable of performing complex functions, taking command of tasks that were previously restricted to desktop computers or other non-mobile systems. Such tasks may include performing searches on the Internet, image acquisition and image analysis and comparison. In addition, mobile devices have physically expanded beyond the boxed design of personal digital assistants. Mobile devices currently include a variety of cell phones, entertainment centers and game consoles as well as wireless robots.

Robotic devices reveal a wealth of new utilities and new possibilities for human computer interaction [1,2]. A robot can be viewed as a physical computer interface that can act in both the physical and virtual worlds. A robot can contribute to a task a set of physical abilities as well as digital capabilities, and distributed computing functions. A robot is capable of pairing audio and visual components of interaction, with physical actions and movements to further enhance the interaction experience.

People often find themselves in situations where they need to acquire specific information about new social or physical surroundings. For example, when entering a new social setting people sometimes need to approach a person whom they never met, or meet a person whom they previously met but fail to recall specific information regarding that person (for example, the person's name). Such shortcomings are sometimes relieved using a partner, spouse or a friend who can retrieve and deliver the required information in a manner that is sociably acceptable. Reliance on information appliances such as PDAs for these situations might not be socially acceptable (for example, trying to search for a person's name or image on a PDA while facing them in the middle of a social event). Further, it is not always possible to have a knowledgeable human companion at every social setting; it might however be possible in the near future to be escorted by a personal robot companion. This is assuming the robot can support socially acceptable interaction and can retrieve information dynamically and efficiently for the user.

This paper describes *Iolaus* (pronounced EE-oh-lus), which is a persistent robot partner that can convey important messages back and help the user navigate through new environments and unfamiliar social settings. In Greek mythology *Iolaus* played crucial role in Hercules' labours and helped him slay the Hyrda [3]. We use *Iolaus*, the obliging partner, as a metaphor motivating our human-robot interaction (HRI) approach to distributed information retrieval.

The *Iolaus* project goal is to develop a robot that serves as a socially acceptable information-supporting partner. The user and the robot are to interactively communicate with each other, with the robot able to sustain sufficient knowledge and awareness of the environment in which it acts. *Iolaus* is designed as a companion and acceptable partner that will behave in a socially unobtrusive manner. *Iolaus* is designed to provide assistance and explore the distributed information realm as naturally as a seeing eye dog negotiates a street corner.

2. Designing Iolaus

Our overall design task was to develop a robot that is capable of serving as an autonomous distributed information retrieval device, which can act in the physical world. Much like a human partner, the robot is able to take in requests from the user, search for the appropriate information, and formulate a response back to the user. The robot can also act in the physical world based on the information it retrieved, for example by pointing to the direction of a person the user is seeking. To allow users to access information quickly and inconspicuously the robot must be designed as a socially acceptable interface.

The information requested from *Iolaus* could vary over a wide range of topics from asking directions to a point of interest, to finding the name of an associate's spouse. Generally, such information can be retrieved either by talking with knowledgeable individuals or by performing a search on the Internet. In a social setting, information should be requested and received in a socially acceptable and non-intrusive manner. Interaction with typical information appliances interfaces or wearable computers can often lead to behaviours that are perceived to be unsociable and often unacceptable.

2.1 Goal

Figure 1. *Iolaus*-a conceptual drawing

Our short-term design goal is to be able to ask *Iolaus* to locate a person in a room in a socially acceptable manner. The person *Iolaus* is asked to seek can either be physically in the room or be represented by a picture. The robot must then decipher the request in such a way that it can search for the information requested. The search for information will include an autonomous gathering of virtual data from a distributed information source (practically by performing a search on the Internet) and gathering of physical data from the robot's visual, audio and tactile sensors. The data gleaned from the robot's sensors is used to compare the images retrieved to those around the room. The robot should compare the visual images in the room to the image it found on the Internet and decide if a match is made. Lastly, the robot will formulate a response back to the user of its findings (for example, by pointing to the right person or walking towards them)

The robot acts as a physical mobile distributed computing device capable of retrieving information about its surroundings. In a sense, the robot is a distributed search engine and email client, capable of deciphering requests, and then formulating a response back to the user.

Our vision for such a distributed information retrieval robotic device would be similar to a robotic parrot that is capable of sitting balanced on the user's shoulder as seen in Figure 1. The close vicinity of the parrot to the user's ear makes it ideal for voice interaction from the user to the robot and vice versa. This is advantageous since both the user and *Iolaus* can communicate discreetly between each other by simply having the user turn their head toward *Iolaus*. The embodiment of *Iolaus* in a parrot shape can allow the robot to interact with the user in various forms: for example, the parrot can physically move its head or wings to show the user where the person is located. The physical movement can also be paired with an audio component where *Iolaus* verbally respond back to the user with some comments or instructions. From a robotic design point of view the physical requirements from the parrot are quite simple as all it needs to do is maintain balance on a shoulder (that is, the robot is not required to walk, navigate corridors, traverse stairways, etc.).

Another possible metaphor for *Iolaus* is an attentive information-seeking dog. This embodiment has the advantages of mobile abilities in the environment on top of the distributed information capabilities. These can be used for instance to help fetch or place something in the physical world. The physical appearance of a dog would also make it more suitable to deal with navigational requests as the robotic dog could in part respond by guiding the user to the desired location, very much like a real seeing-eye dog.

2.2 Query Request

Iolaus must be made aware of a query request. This can be done by issuing voice commands to the robot, in a similar manner to verbal queries between two people. The verbal interaction between the robot and the user then allows for a high level and more natural means of interaction.

2.3 Information Retrieval

Iolaus should be capable of autonomously retrieving information. It should not be dependent on any external computational source other than what is contained within itself. Therefore, the robot is not tethered to any devices and is capable of being completely mobile. The task of information retrieval then is delegated completely to the robot. First, *Iolaus* must perform a search to find the correct answer for the user query. In the task of attempting to correctly identify a person in a room, the robot must be able to find the correct image of the person. Such an image can be obtained from the Internet through various web engine searches and the robot should be capable of going online, performing the search and obtaining the results from the search engine. Second, the robot must gather information from its surroundings through its visual sensors. Once it has successfully obtained an image from the Internet, the robot must look around the room and match the retrieved image with one that it sees through its vision system.

2.4 Formulating a Response

Finally, after the information retrieval phase, *Iolaus* should formulate its response into an appropriate form and deliver it to the user. This can be done in a number of different ways depending on the situation and application. For example, the robot can send navigational directions back to the user or physically guide the user to the location of the desired person using visual, verbal or physical interaction techniques.

2.5 Physical Interaction

In order for *Iolaus* to be an acceptable and useful human-robot interface it should be based on intuitive interaction techniques. The robot physical state must afford and reflect its current virtual status. For example, if the robot is in the middle of searching for an image on the Internet, it should physically demonstrate that it is busy with a task and cannot be bothered. When *Iolaus* is waiting for the next user request it should physically reflect that it is willing to serve a new query. The robotic interface should not be restricted to visual and audio stimulation but rather enhance the experience by controlling and using the robot's physical state as a valid output and display tool.

3. Implementation

To formulate a robot capable of fulfilling the design tasks outlined, we used the Sony Aibo ERS-7M2 for *Iolaus's* prototype. The Aibo is an autonomous robotic dog that has a face plate LED panel, a 350K pixel image sensor, a speaker capable of producing MIDI sound, stereo microphone, head/paw/body sensors as well as IEEE 802.11b Wireless LAN capabilities.

Our Aibo-based *Iolaus* was implemented using OPEN-R: a C++ SDE for the Aibo robot dog available for free from Sony [4]. Aibo's OPEN-R supports coding most of the robot-dog functions as well as its IPv4 Internet protocol and TCP (Transmission Control Protocol). We used these capabilities to program *Iolaus* to retrieve emails and perform searches on the Internet.

Since we are looking for multimedia content, we decided that our Aibo-based *Iolaus* prototype will use a search engine that returns a good array of pictures such as www.picsearch.com [5].

3.1 Iolaus Prototype

The current early prototype of *Iolaus* is capable of performing a simplified but fully functional experiment that demonstrates the potential and usability of the new robotic interface as well as its functionalities in a distributed setting. Our current implementation allows the Aibo to autonomously search and find an image on the Internet based on a user query. The requests from the user are sent via email to Aibo's private email account (Figure 2). Our Aibo-based *Iolaus* checks its emails regularly for new requests from users. Based on the user's request for information, Aibo will parse through the email, create a search engine query, connect to the Internet and perform a search on a public search engine to find an appropriate picture answering the user's query. After it has successfully extracted the results, Aibo will then package its response into an email and reply back to the user with the link to the online picture. Through this cycle, Aibo will perform various physical/visual movements that allow the user to be aware of its current status.

3.2 Distributed Information Retrieval

The act of connecting to the Internet and searching for an image is very similar to the distributed information retrieval task as outlined in the design plan.

Our Aibo-based *Iolaus* is capable of receiving and sending emails and so contains a fully functional independent email client. Using any email account with POP/SMTP capabilities, Aibo is able to consistently monitor and retrieve emails it receives. The Aibo-based *Iolaus* is also able to delete email messages that have been processed, so once Aibo has tried to search for a user query it will remove that email message from its inbox. This is one way that Aibo can manage its email messages and ensure that its email account doesn't overfill. Also, if the Aibo-based *Iolaus* sees that the subject heading on the email is not something it recognizes, it will automatically remove the message from the queue.

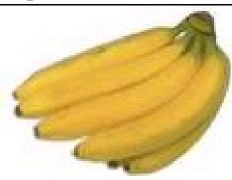
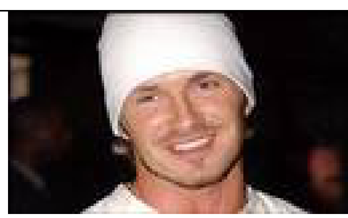
3.3 Robotic Multimedia Human Computer Interface

The key to *Iolaus* is to provide the user with a multimedia robotic interface that is natural to interact with in a social setting. This was achieved in the current prototype by having Aibo physically demonstrate its status. Our Aibo-based *Iolaus* assumes different physical doggish postures and movements reflecting its progress and virtual state (Figures 3,4).

4. Evaluation

To evaluate the effectiveness of the prototype in the original design task we present several complete request/response cycles that were tested using our current Aibo-based *Iolaus* prototype.

Table 1. Other examples of 'Go Fetch' requests.

Go Fetch:	Response back
Banana	
Aibo	
Beckham	
Apple	

The requests made to Aibo are in the form of an email message sent to an email account created specifically for Aibo (*Iolaus* registered to a Yahoo email account). Currently, Aibo will only accept email messages in a specific format. The email message itself must contain the phrase 'Aibo, go fetch:', followed by the name of the item the user is looking for. A typical request may be in the form of: "Aibo, go fetch: bananas.". The message must conclude with an acknowledgement of Aibo's efforts by including, "Thanks, " at the bottom of the message. Figure 2 shows a request made from a user email to our Aibo-based *Iolaus*.

From:	"Ellen Lau" <ellenlau.28@gmail.com>
To:	iolaus_aibo@yahoo.ca
Subject:	Ask Aibo

```
Aibo, go fetch: kogepan.
Get back 10 results.
Thanks,
```

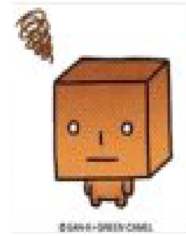


Figure 2. An email request sent to Aibo (left), and the image retrieved back (right).

Autonomously parsing through the email, Aibo will isolate out the item to search for and formulate a well-formed HTTP request [6]. A typical request may be: GET / search.cgi?q=kogepan HTTP/1.0\r\n. In this request, Aibo has isolated the item "kogepan" from the emailed request and inserted into the proper HTTP request form. The searches for images are done through the image search engine on Picsearch.com. The letters after the GET command in the HTTP request simply tell Picsearch that an image search is going to take place.

After the HTTP request is sent and the server returns a response, Aibo will proceed to parse through the web page looking for the first image link on the result page. The link will be the first answer that Aibo provides back to the user. In our example, the image link that Aibo has found (Figure 2) is:
http://images.picsearch.com/is?5261000945372.

Finally, Aibo creates a return email back to the user indicating that he has found an image link answering the user query. The message also contains the link so that the user can access the image.

More examples of queries and resulting images are presented in Table 1. As seen, there are scenarios where the images retrieved can be informative, and in other cases there is a need for further processing of the results. For instance, in the case involving the 'fetch apple' request, it could very well be that the user was looking for the fruit, not a computer system. If the user is not satisfied with *Iolaus's* response the query can be easily repeated and extended.

The Aibo-based *Iolaus* physical posture and actions provide the user with insight into the interface state. When Aibo is about to search on the Internet, it performs a 'stretch' by flexing its legs out as shown in Figure 3.

This appears as if the robot dog is physically getting ready to perform a complicated task.

Once a response email is sent back to the user, the dog sits very much like a watchdog. Our Aibo-based *Iolaus's* head sways from side to side (Figure 4) as if looking for an interesting game to play while it searches and waits for user's next emailed request. As an added method of physically conveying its virtual state, Aibo also flashes its LED lights on its head, back and face when it is in the process of sending and retrieving emails. The lights will turn off when Aibo is awaiting a user request.

Figure 3. Aibo 'stretching', indicating its readiness to search the Internet

Figure 4. Aibo's head turning waiting for a new query.

5. Related Work

Sociable robots need to perceive, recognize and interpret the behaviour of humans through multiple modalities including vision, hearing and touch. [1,2]. A social robot should be able to naturally interact with humans and participate in human society in ways that will be perceived as natural and socially acceptable [1,7].

Robotic emotions can play a key role in the perceived quality of a human-robot interface [8]. The robot can use emotions as tools to understand and convey implicit messages to and from the user. In addition, *Iolaus* can use synthetic emotions to elicit specific responses from the user. Emotions are crucial in formatting user impressions for reactions in different circumstances [8].

At first glance, the robot must possess qualities that make it appealing and socially acceptable and its appearance must match its function. A robot's appearance should provide clues to its purpose much the same as an interface should afford its behaviour and function [9]. It was demonstrated that users systematically preferred robots with human likeness features that match the sociability levels required in specific tasks [1,10]. Following, it is then easy to conceive *Iolaus* as a parrot perched on the user's shoulder or a robot puppy that guides users to a desired location similar to a Seeing Eye dog.

Iolaus needs to provide meaningful information back to the user. Since an autonomous robot has limited resources onboard, it will need to search for external information. This source could be a specialized database or the Internet. For multimedia content, very similar to textual queries, more advanced analysis should be performed on the query results to gain meaningful information. Previous work on image clustering [11,12] and image retrieval [13] already enable more advanced analysis of image results, and relevance feedback techniques which are suitable to human-robot interaction can further enhance its significance.

6. Conclusion and Future Work

People often need to go to new places, meeting new people and perform various social functions in new social settings. Often they do not have a companion that can accompany them on these new tasks. We design *Iolaus* to serve as a robot companion that can perform the function of a knowledgeable human partner without the social awkwardness associated with using various information appliances. *Iolaus* can fill requests from the user regarding their surroundings and interact in such a way that conforms to social norms. *Iolaus* would ideally be placed on the user's shoulder, very much like a parrot but can also assume other forms such as an attentive dog.

We have presented an early prototype of the project, where *Iolaus* is simulated using the Sony Aibo robot dog.

The Aibo's task is very similar to what *Iolaus* is required to do. The user poses a request query for Aibo to find an image of an object. The query is directed in the form of an email sent to Aibo's email account. *Iolaus* checks its emails, parses new ones, and forms the request item into an HTTP request. This is done so that Aibo can autonomously take the request and wirelessly access the Internet via a search engine to retrieve a set of possible results. Once the results are returned, The Aibo-based *Iolaus* determines the appropriate answer by isolating an image link on the page and sends a response email back to the user.

We are currently working on ways to expand the human robot interaction with our implementation of *Iolaus*. Some of the tasks include:

- Developing a voice interface with Aibo so that requests and responses can be made via voice commands. Ultimately, we would like *Iolaus* to fully support voice interaction as well as the ability to communicate and understand basic emotions.
- Adding a physical interaction technique that will allow the user to cycle between multiple query results. For example, use the sensors on Aibo's paws and back to browse through the next image in the result set.
- Ultimately, *Iolaus* should be able to interact via physical means. This could mean gesturing of appendages or locomotion to a desired location. Whatever the means of interface *Iolaus* offers, it should be an amalgam of various multimedia outlets that best portray its response as natural for its social setting.
- Preprocessing the multimedia results in a more informative manner by clustering or analyzing the images. This will later enable for instance searching similar objects in the environment using the Aibo's visual system.
- Filtering system for better result matching. A large portion of what is received back from Aibo is dependent on the effectiveness of the search engine that is used. One way to retrieve more accurate results could be to provide Aibo with more information for the query. Again though, this solution is highly dependable on the effectiveness of the images search engine that is used.

While the experimentation using the Aibo is very much a simplified form of *Iolaus's* planned functionality, there are important components that the Aibo prototype can help accomplish such as the wireless and dynamic retrieval of multimedia information, which is accessed remotely. The success of these components will help advance the goal of realizing the completion of *Iolaus*.

Acknowledgments

This work was partially funded by the Canadian Natural Sciences and Engineering Research Council (NSERC) and a University of Calgary start up grant.

References

[1] C. Breazeal, *Designing Sociable Robots,* The MIT Press. 2002.

[2] C. Breazeal, A. Brooks, J. Gray, M. Hancher, J. McBean, D. Stiehl, J. Strickon, "Interactive Theatre" in *Communications of the ACM,* July 2003, Vol 46, No. 7, http://robotic.media.mit.edu/Papers/Breazeal-etal-CACM03.pdf

[3] C. House, Hercules the hero: understanding the myth, 1998 http://www.yale.edu/ynhti/curriculum/units/1998/2/98.02.06.x.html

[4] SONY AIBO Software Development Environment, http://openr.aibo.com/openr/eng/index.php4

[5] Picsearch, http://www.picsearch.com

[6] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee "Hypertext Transfer Protocol -- HTTP/1.1" *The Internet Society.* June 1999. http://www.w3.org/Protocols/rfc2616/rfc2616.html

[7] T. Kanda, H. Ishiguro, T. Ishida, "Psychological Analysis on Human-Robot Interaction," in *Proceedings. of IEEE Int. Conf. On Robotics and Automation*, 2001, http://www.lab7.kuis.kyoto-u.ac.jp/services/publications/01/01conf04.pdf.

[8] D. A. Norman, *Emotional design: Why we love (or hate) everyday things.* Basic Books, New York, 2004.

[9] D.A. Norman, *The Design of Everyday Things.* Basic Books, New York, 1988.

[10] J. Goetz, S. Kiesler, A. Powers, "Matching Robot Appearance and Behavior to Tasks to Improve Human Robot Cooperation," in *University of California, Berkeley, Carnegie Mellon University.* 2003. http://www.contrib.andrew.cmu.edu/~rgockley/papers/AppearanceAndBehavior.pdf

[11] S. Krishnamachari and M. Abdel-Mottaleb, Hierarchical clustering algorithm for fast image retrieval. *In SPIE Conference on Storage and Retrieval for Image and Video databases VII*, pp. 427-435, San-Jose, CA, Jan 1999.

[12] Kobus Barnard, Pinar Duygulu, and David Forsyth, Clustering Art, *In Computer Vision and Pattern Recognition,* pp. II:434-439, 2001.

[13] Y. Rui, T. S. Huang, and S.-F. Chang, Image retrieval: Current techniques, promising directions, and open issues *Journal of Visual Communication and Image Representation,* 10, 39-62. 1999.

Handling the Occlutions in Fractal Face Recognition and Retrieval

Michele Nappi, Daniel Riccio, Genny Tortora
Dipartimento di Matematica e Informatica
Università di Salerno,
84084 Fisciano (SA), Italy
{mnappi, driccio, tortora}@unisa.it

Abstract

Facial image analysis is very useful in many applications such as video compression, talking heads, or biometrics. During the last few years, many algorithms have been proposed in particular for face recognition using classical 2-D images. However, it is necessary to deal with occlusions when the subject is wearing sunglasses, scarves and such. In this paper, a new face recognition method is proposed. The method is based on IFS (Iterated Function Systems) theory. One advantage is that the information used for the indexing and recognition task can be made local, and this makes the method more robust to possible occlusions. The distribution of similarities in the face image is exploited as a signature for the identity of the subject. The amount of information provided by each component of the face image has been assessed, first independently and then jointly. At last, results underline that the system significantly outperforms the existing approaches in the state of the art.

1 Introduction

Recently, security-related problems are drawing much of the researchers' attention. Particularly, the problem of authenticating people is an interesting and stimulating research field. Many anatomical features, such as fingerprints, hand-shape or eye iris have been studied up to date. It has been observed that two main characteristics enact the success of a biometry: reliability and people acceptance. Indeed, iris recognition represents the most reliable approach, but it is too intrusive to be widely accepted. On the contrary, fingerprints are easily applicable, but they cannot be used with non consenting people. Therefore, face recognition is emerging as a compromise between acceptance and reliability.

The recognition rate of face-based biometric systems cannot be compared to that of iris and fingerprint based systems. However, the ease in snapshot and video capturing make this method effective also when the subject does not collaborate or ignores that recognition is going on. Furthermore, the wide range of commercial and law enforcement applications supported by the availability of feasible technologies favor the success of the face biometric. Several methods have been devised to perform the face recognition, dealing with many of the typical issues in face recognition, such as variation in expression, lighting, pose and acquisition time, but none of them is free from limitations. Eigenfaces are a fast, simple and practical technique, but they are not invariant with respect to changes in illumination, pose and scale.

There are few works about the problem of occlusions, such as [3, 5], based on probabilistic approaches or neural networks.

The main contribution of this paper is then the application of a fractal based technique, namely OFF (Handling the Occlutions in Fractal Face Recognition and Retrieval), that offers a possible solution to the face recognition problem in the presence of synthetic and natural occlusions, such as global pixel wise errors and rectangular occlusions or sunglasses and scarves. The proposed strategy applies IFS (Iterated Function Systems), largely used in image compression and indexing [1]. The affine transformations are used in order to characterize the self-similarity of a face image, extracting a compact feature vector with high discriminant power, indeed OFF significantly outperforms other approaches in literature.

The remainder of the paper is organized as follows. In Section 2 shows in more detail the feature extraction process. In particular, Subsection 2.1 describes the structure of the feature vector while Subsection 2.2 introduces a new "distance" function needed in order to make comparisons among models computed for the input face images. In Section 3, a concise description of the measures and databases used during tests is given together with results. Lastly, the paper ends with a few concluding remarks in Section 4.

2 The Method

In order to make method presented in this paper robust with respect to likely occlusions, the feature extraction process is made local to the region, of interest, defined as the union of four main areas, which are left eye, right eye, nose and mouth (Fig. 1 (a)). For each of these areas, a set of fiducial point is extracted and the average approximation error is computed, so that point locations and approximation errors represent the signature for the face (Fig. 1 (b)).

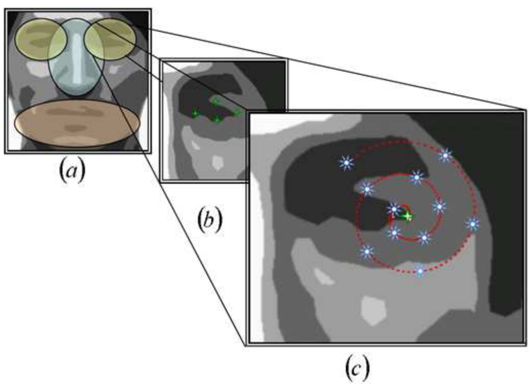


Figure 1. Region of interest (a), location of the entry points on the face image (b)and the Archimede's spiral centered in an entry point (c).

The first problem to be solved is locating the face in the picture. The detection of a face is semi-automatical. The center of the eyes and baseline of the nose are manually selected, extracting a face region of size proportional to the distance among these three points. The face region extracted from the input image is normalized to 256×256 pixels. Nothing has been done for the original warping of the input images, which also can affect the recognition rate. Once the segmentation of the face is done, the face objects are independently indexed by means of the IFS systems as separate region of interest. For each region of interest (eyes, nose or mouth) a set of fixed points $P = \{P_1, P_2, \ldots, P_n\}$, called *entry points* is considered.

For each $P_i = (x_i, y_i) \in P$, OFF extracts the corresponding range R_{x_i,y_i}, whose upper-left corner falls in the position P_i. It searches for the first n best fitting domains with respect to an affine transformation, as detailed in [8]. In order to render the method more robust with respect to small shifts around the entry position (x_i, y_i) 18 nearest neighbors of the current entry points are considered. Starting from the current entry point, all centroids are computed by means of the above algorithm, then with a spiral visit centered in the current entry point, 8 neighbors are considered, as shown in Fig. 1 (c). The neighbors are looked for on an Archimede's spiral $\rho = a \cdot \theta$, where ρ is the distance from the center to the tracing point, θ is the angular distance covered and a is a fixed constant. With the spiral visit, the higher the distance from the entry point, the less useful is its information. This can be explained considering that the further the neighbor is from the current entry point, the less their similarity to the current range.

In order to use the information about the distribution of similarities brought out during the indexing phase, range/domain relations have to be organized so that comparisons are possible. In this case, the domains are organized in a set of clusters $C = \{C_1, C_2, \ldots, C_m\}$, each represented by its centroid c_i, and the centroids are stored in memory as a list. Each centroid object in the list includes its spatial coordinates, computed as the mean of the coordinates of domains to the cluster, as well as the average approximation error between the domain and the prefixed block:

$$C_x = \frac{\sum_{D \in C} D_x}{|C|}, \quad C_y = \frac{\sum_{D \in C} D_y}{|C|} \tag{1}$$

$$C_{\text{err}} = \frac{\sum_{D \in C} \sqrt{\sum_{i=1}^{|PB|} \left(PB(i) - (\alpha D(i) + \beta)\right)^2}}{|C|}. \tag{2}$$

At first the list of centroids is empty; then, starting from the first entry point, the corresponding range is extracted and the n best fitting domain are searched. The domains are inserted in the clusters one at a time. Initially the list of centroids is empty, then a new cluster is created and its centroid has the same coordinates of the inserted domain. The second domain fitting the current range is extracted by means of the IFS transformation and has to be inserted in the correct cluster. OFF scans the list of centroids searching for the centroid with minimum distance. If no suitable cluster is found, a new cluster is created and added to the list, while the corresponding centroid has the same coordinates and approximation error of the inserted domain. On the contrary, if there is a cluster whose centroid is not further than a fixed threshold ϵ from the domain to be inserted, OFF tries to update the cluster with the new domain. The new coordinates are computed for the centroid according to (1) and (2) but taking into account also the new domain being inserted. After updating the coordinates, a check is made where there is a domain in the cluster whose distance from the centroid is greater than the fixed threshold ϵ. In this case, the updating operation is canceled, and a new cluster is created with the same coordinates and approximation error of the inserted domain.

2.1 Feature Vector Structure

Once the list of centroids has been computed, it has to be rearranged so that a distance function can be defined for the comparisons to be performed later. The ideal way to compare two face images would be to match the respective approximation errors of corresponding centroids in the two images. Since each centroid in the list consists of two coordinates $C(x,y)$, the nearness of two centroids can be estimated in a simpler way using Euclidean norm.

Let L_1 be a list of centroids of length $n = |L_1|$ to be compared with another list L_2, of length $m = |L_2|$. An effective manner to compare the centroids in L_1 and L_2 is representing the centroids' spatial location with Peano keys. From the literature, it is known that Peano keys are useful in indexing tasks because they map a 2D space into a 1D space, while preserving most of the spatial information in the original data. Given a centroid $C(x,y)$, the correspondent Peano key a_C is computed interleaving bits of x and y, from the less significant digit to the most significant. The Peano keys so computed are then sorted. This can be done in linear time $O(n)$ with Radix Sort.

When comparing L_1 and L_2, a time $O(m)$ is spent searching for the centroid $C^j_{L_2}$ in L_2 nearest to the first centroid $C^1_{L_1}$ in L_1, keeping memory of j. It can be observed that the location of the next centroid in L_2 nearest to $C^k_{L_1}$ falls not so far from the position j—indeed, it is about $j+c$, where it has been found experimentally that $0 \le c \le 10$, and j is the position in L_2 of the centroid nearest to $C^{k-1}_{L_1}$ in L_1, with $k > 1$. It can be deduced that for each centroid in L_1, only c centroids in L_2 have to be tested. The overall complexity of the comparisons is then $O(n + c \cdot m) = O(n+m)$—linear, since c is a constant. Low-complexity comparisons are crucial, considering that in a huge database of face images, millions of images might have to be tested.

2.2 Definition of the Distance Function

This section defines the distance function used when comparing two feature vectors. The domain of this function consists of 2D vectors $S \in \Re^2$, where $(a,b) \in S$. The symbol a represents a Peano key obtained, while the real value b represents the average of the approximation error for the centroid centered in $a = P(x,y)$. Given 2 vectors $S, T \in \Re^2$, the operator $\Psi(S,T)$ is defined as follows.

$$\Psi_i(S,T) = \left| b_T^{\mu(S,T)} - b_S^i \right| \tag{3}$$

with

$$\mu(S,T) = \min_j \left\| a_T^j - a_S^i \right\|_2 ,$$

that is, $\mu(S,T)$ represents the index in T of the point $a_T^j = P_1(x_1,y_1)$ nearest to the point $a_S^i = P_2(x_2,y_2)$ in S.

For each item $a_S = P(x,y) \in S$, a search is performed for the nearest item $a_T = P(x,y) \in T$ according to Euclidean norm $|P_S - P_T|_2$, and the quantity $|b_S - b_T|$ is computed. This quantity represents the absolute difference between the approximation errors corresponding to the nearest points $P_S \in S$ and $P_T \in T$.

Lastly, the values of $\Psi_i(S,T)$ are summed for all i. In order to make the distance function more robust to partial occlusions, it can be noted that if $\Psi_i(S,T)$ is too large, it does not supply much useful information, and this can be interpreted as a sign of possible occlusion.

More precisely, if $\bar{m} = \frac{1}{|S|} \cdot \sum_{i=1}^{|S|} \Psi_i(S,T)$ is the average value of Ψ over S, it turns out that only the values of $\Psi_i(S,T)$ ranging from 0 to $2\bar{m}$ provide useful information. Therefore, a threshold is applied to cut all values above $2\bar{m}$, leaving smaller values untouched. This is done by means of the following function:

$$\Delta(S,T) = \frac{1}{\left|\tilde{S}\right|} \sum_i^{|S|} \gamma_S^i \cdot \Psi_i(S,T) + \frac{1}{\left|\tilde{T}\right|} \sum_i^{|T|} \gamma_T^i \cdot \Psi_i(T,S), \tag{4}$$

where

$$\gamma_S^i = \frac{(S(i) - 2 \cdot E[S]) - |(S(i) - 2 \cdot E[S])|}{2 \cdot (S(i) - 2 \cdot E[S])}$$

and

$$\tilde{S} = \left\{ (a_i, b_i) \in S | \gamma_S^i \neq 0 \right\}.$$

3 Experimental Results

There are several standard databases used to assess the performance of algorithms in the field of face authentication. Two of the most used face databases are FERET [7] and AR Faces [6]. The FERET facial database has become the de facto standard for evaluating face recognition technologies and consists of 13539 facial images corresponding to 1565 subject of different gender, ethnicity and age.

The AR Faces [6] was created by Aleix M. Martinez and Robert Benavente at the Computer Vision Center (CVC). It contains over 4000 color images corresponding to 126 people's faces (70 men and 56 women). Images feature frontal view faces with different facial expressions, illumination conditions, and occlusions (sun glasses and scarf). The pictures were taken at the CVC under strictly controlled conditions. No restrictions on wear (clothes, glasses, etc.), make-up, hair style, etc. were imposed to participants. Each person participated in two sessions, separated by two weeks time. The same pictures were taken in both sessions.

The measure used to assess the performance of OFF in solving the identification problem is the recognition rate, defined as the ratio between the number of probes correctly identified in the first response and the cardinality of the whole probe set.

Since the main goal when designing OFF was to be able to deal with partial occlusions, most experiments have been performed with the aim of assessing performance in the case of authentication with partial occlusions. For these experiments, the performance of OFF has been compared with the results obtained in analogous conditions by a probabilistic approach proposed by Martinez [5], referred to in this paper as PAO and a neural network based approach described by Kurita [3], namely NNFR I/II.

PAO divides each face image into k different areas, each of which is represented by a Gaussian distribution accounting for the positioning error. The use of Gaussian distributions allows the mean feature vector and covariance matrix to be calculated for every local subspace, while the probability of a given match can be calculated as the sum of all k Mahalanobis distances.

Kurita *et al.* proposed in [3] a method that also reconstructs the occluded part of the face and detects the occluded regions in the input image, by means of an auto-associative neural network. At first the network is trained on the non-occluded images in normal conditions, while during the testing the original face can be reconstructed by replacing occluded regions with the recalled pixels. In this case we chosen the original neural network based approach reported in [3] (NNFR I) and the improved version proposed by Kurita *et al.* without the recursive data reconstraction process (NNFR II).

In the first experiment, OFF has been compared with the PAO approaches in the case of synthetic square occlusions. A subset of 50 subjects from the AR face database has been considered. For each face, the non-occluded neutral expression is used for training the system (gallery), while occluded neutral, angry and smile images are used for testing (probe). Synthetic square occlusions of size $p \times p$ have been introduced in the probe images, with p varying from a minimum of 5 to a maximum of 50. In particular, for each value of p, a $p \times p$ blank square is randomly placed into the image 100 times. The averaged results are shown in Fig. 2. For large values of p, a gap of about 10% in the Recognition Rate of the two methods can be seen. This confirms the robustness of OFF with respect to the synthetic occlusions. Moreover, the PAO approach divides the face image in six elliptical regions, while OFF uses only four rectangular zones. A smaller number of independent regions means that higher robustness of the method is required, since the occlusions can affect more regions simultaneously.

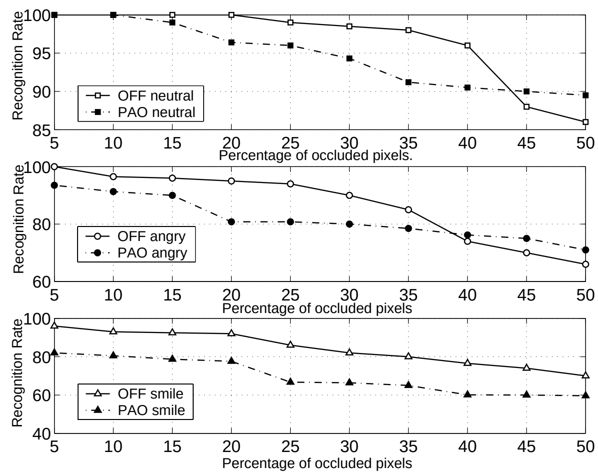


Figure 2. Performance of OFF when synthetic rectangular occlusion occur; comparison with Martinez' PAO algorithm.

The second experiment is for testing OFF in the case of natural occlusions, such sunglasses. In particular the performance of the fractal based method is compared to that of a neural network based approach NNFR. The training data set consisted of ninety three 18×25 8-bits images (3 for each of the 31 persons), while the sunglass data set consists of real images of the same person as the training data but wearing sunglasses (31 images). Table 1 underlines that OFF outperforms NNFR I/II, if no reconstruction is allowed 64.5% and 77.4%.

OFF	NNFR I	NNFR II	NNFR III
82.04%	64.5%	77.4%	87.10%

Table 1. Comparison between OFF's and Kurita's algorithms on the AR database.

This experiment underlines that OFF allows a more localized and compact description of the face with respect to neural network based approaches, when natural occlusions occur, because unuseful informations provided by the occluded zones do not affect significantly the global description of the face.

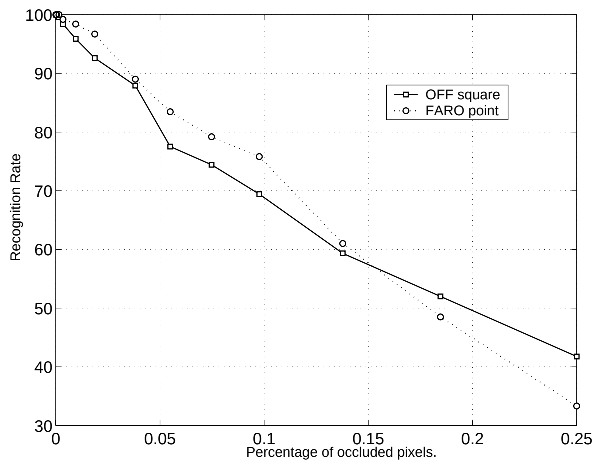


Figure 3. Performance of OFF on the FERET face database with square and pointwise synthetic occlusions.

In the last experiment OFF has been tested on a subset of the FERET face database. The gallery set consists of 400 images, two for each of the 100 different subjects. On the contrary, the probe set contains 100 images, one image for each subject in the gallery. Two different kind of synthetic occlusions have been considered: square and pointwise occlusions. In the first case, a black $p \times p$ square (with p the side lenght of the square) has been placed on the probe image, that is then used as a query to the system. On the contrary, in the second case, exactly $p \times p$ pixels on the probe image have been flipped to black before than queryng the system with this image. Fig. 3 reports the recognition rate with respect to the percentage of the occluded pixels ($p^2/256^2$) in the probe image. This experiment make clear that OFF is more robust with respect to pointwise occlusions than the square ones, when p is relatively small. By the way, it has been observed that OFF is robust to moderate random noise for not very large values of p. On the contrary, for larger values of p it shows better performances when the occluded part is concentrated in a single region of the face.

At last Fig. 4 shows time spent searching for a face on a 1 to 400 image database, when the match between two faces is carried out comparing the approximation error of each centroid in the former face with the approximation error of the nearest centroid in the latter (OFF normal), and when the Δ (Subsection 2.2) distance is used.

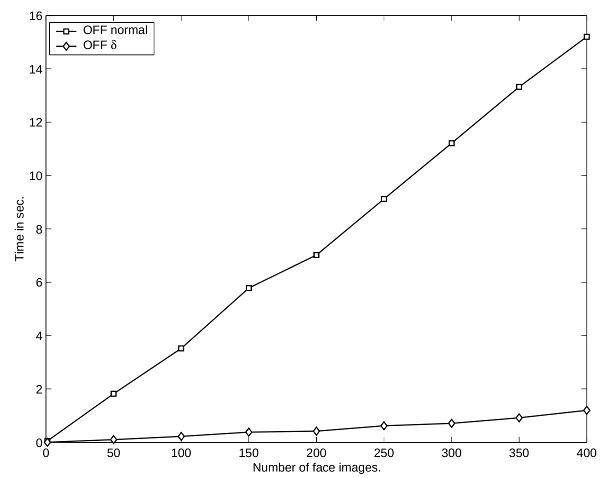


Figure 4. OFF efficiency with full comparisons, tree organization and Δ distance.

4 Conclusions

The interest of researchers for face recognition is firmly increasing recently, so that several solution have been proposed up to date. However, most recent methods only deal with changes in expression and illumination conditions, and yield poorer recognition with the case of synthetic and natural occlusions. In particular, the literature about fractal approaches to face recognition appears to be somewhat underdeveloped. This paper proposes a new fractal based approach called OFF. It is shown how IFS transformations can be used as to to provide a good signature for face images, useful for face recognition. The indexing process has been made local, and a new metric is provided in order to deal with partial occlusions. Experimental results show the robustness of OFF with respect to synthetic occlusions (black squares randomly localized on the image) and natural occlusions such as scarfs and sunglasses. OFF has been compared with a neural network based approach and with the PAO method in order to assess its performance on natural and synthetic occlusions. Further work might address the problem of automatic and finer location of the face objects (eyes, nose and mouth), as well as an increase in robustness to changes in illumination conditions.

References

[1] R. Distasi, M. Nappi, M. Tucci "FIRE: Fractal Indexing with Robust Extensions for Image Databases" , in *IEEE Transactions on Image Processing*, vol. 12, Issue: 3, pp. 373–384, March 2003.

[2] A. K. Jain, A. Ross and S. Prabhakar "An Introduction to Biometric Recognition" , in *IEEE Transactions on Circuits and Systems for Video technology*, vol. 14, no. 1, pp. 4–20, January 2004.

[3] T. Kurita, M. Pic and T. Takahashi "Recognition and Detection of Occluded Faces by a Neural Network Classifier with Recursive Data Reconstruction" , in *Proceedings of the Advanced Video and Signal Based Surveillance (AVSS'03)*, vol. –, no. -, pp. – – –, – 2003.

[4] S. Lawrence, C. L. Giles, A. C. Tsoi and A. D. Back "Face recognition: A convolutional neural-network approach" , in *IEEE Transactions on Neural Networks*, pp. 98–113, 1997.

[5] A. M. Martnez "Recognizing Imprecisely Localized, Partially Occluded and Expression Variant Faces from a Single Sample per Class" , in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 748–763, June 2002.

[6] A. M. Martnez "The AR face database" , CVC Tech. Report no. 24, 1998

[7] P. J. Phillips, H. Moon, S. Rizvi and P. Rauss "The FERET evaluation methodology for facerecognition algorithms" , in *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, October 2000.

[8] Riccio D., Nappi M. "Defering range/domain comparisons in fractal image compression" , in *Proceedings 12th International Conference on Image Analysis and Processing*, vol. 1, pp. 412–417, September 2003.

[9] W. Zhao, R. Chellapa, P. J. Phillips and A. Rosen-Feld "Face Recognition: A Literature Survey" , in *ACM Computing Surveys*, vol. 35, no. 4, pp. 399–458, December 2003.

Beard Tolerant Face Recognition Based on 3D Geometry and Color Texture

Andrea F. Abate, Stefano Ricciardi, Gabriele Sabatino, Maurizio Tucci
Università degli Studi di Salerno, Dipartimento di Matematica e Informatica
84084 Fisciano (Salerno), Italy
{abate, sricciardi, gsabatino, mtucci}@unisa.it

Abstract— **Face is recently becoming an ever more accepted biometric in many applications requiring identity recognition or authentication without physical contact. Most ongoing researches are mainly focused on increasing recognition accuracy and robustness to lighting or posing variations, but other relevant issues related to facial characterization, such as the presence/absence of beard, still need to be effectively addressed. This paper presents a face recognition method which operates on both 3D geometry and color texture to improve recognition reliability in case of subjects with beard. Indeed, the proposed method represents 3D face geometry through a normal map and flesh color by 2D texture map. This approach allows to compare any two faces simply through a comparison of their corresponding normal maps, using the flesh color map to mask non-skin features. A weighting mask, automatically generated for each subject using a set of expression variations, improves the robustness to a broad range of facial expressions. First results show the effectiveness of this simple and fast method on a database of 3D faces featuring different genders, ages and expressions.**

I. INTRODUCTION

Among the various biometric identifiers currently available (fingerprint, face, ear, iris, retina, gait, signature, etc.) face is widely considered one of the more acceptable as it does not require any contact with the sensor surface, nevertheless it allows good recognition performances.

The early researches on 3D face recognition were conducted over a decade ago as reported from Bowyer et al. [1] in their recent survey on this topic and many different approaches have been developed over time to address this challenging task. The comparison between 3D face representations can be performed according to different techniques and strategies. Indeed, while some approaches work best on neutral faces, i.e. faces showing a standard "relaxed" expression, others try to cope with shape variations due to random facial expressions. Some methods are based on feature extraction and they represent face curvature and metric size properties as a point in feature space, whose distance to other points (faces) measures facial similarity [2].

Various extensions of 2D face recognition techniques to the range images have been proposed, such as those based on eigenface [3] or Hausdorff distance matching [4]. Other works compare faces through a spherical correlation of their Extended Gaussian Image [5], or through Principal Component Analysis (PCA) [6], or even measure the distance between any two 3D facial surfaces by the Iterative Closest Point (ICP) method [7]. To increase recognition rate in case of expression variations Bronstein et al. [8] presented a new method based on a bending invariant canonical representation, the canonical image, that models deformations resulting from facial expression and pose variations which are supposed to be isometric transformations. Multimodal approaches combine 2D (intensity or color) and 3D (range images or geometry) facial data to improve recognition accuracy and/or robustness over both 2D and 3D techniques alone. Chang et al. [9] performs separately the PCA on the intensity and range images, combining the results obtained from both strategies to get a more reliable recognition. Tsalakanidou proposed [10] an approach to integrate depth data and intensity image exploiting embedded hidden markov model. Papatheodorou and Ruecker [11] presented a 4D registration method based on Iterative Closest Point (ICP), augmented with texture data. The proposed metric exploits Euclidean distance between 4D points (each featuring three spatial coordinates plus corresponding texel intensity). Other authors combine 3D and 2D similarity scores obtained comparing 3D and 2D profiles [12], or extract a feature vector combining Gabor filter responses in 2D and point signatures in 3D [13].

We present a face recognition method aimed to biometric applications and based on 3D geometry and color texture. This novel approach features high recognition precision, good robustness to a broad range of expression variations and to the presence/absence of beard and/or moustaches. A low one-to-one comparison time allows to efficiently apply this method even on very large gallery.

This paper is organized as follows. In section II. the proposed methodology is presented in detail. In section III. experimental results are shown and briefly discussed. The paper concludes in section IV. showing directions for future research.

II. A NORMAL MAP BASED APPROACH TO FACE RECOGNITION

The proposed method is based on normal map [14] to represent and compare faces. A normal map is simply an RGB color image providing a bidimensional representation of a 3D surface, in which each normal to each polygon of a given mesh is represented by an RGB color pixel. More precisely, each pixel in a normal map encodes the three scalar components of the normal to a particular polygon in the mesh surface using the three RGB channels usually associated to a color image. This bidimensional representation of original face geometry still retains spatial relationships between facial features. Figure 1. shows the method's workflow which could be briefly exposed as follows. After a face has been enrolled, its surface mesh is generated and the color texture is mapped onto it, projecting the texture's 2D coordinates onto the 3D surface (through a spherical or cylindrical transformation). The corresponding normal map is computed sampling surface's normals. The comparison between any two faces is therefore performed computing a new map called the *Difference Map*, which is a gray scale image obtained subtracting pixel by pixel the two corresponding normal maps. Eventual beard covered facial regions are masked according to their relevance through a gray scale *Flesh Mask*. Finally, to cope with unwanted facial expression the difference map is multiplied by a previously built expression weighting mask. The whole recognition process is discussed in depth in the following subsections.

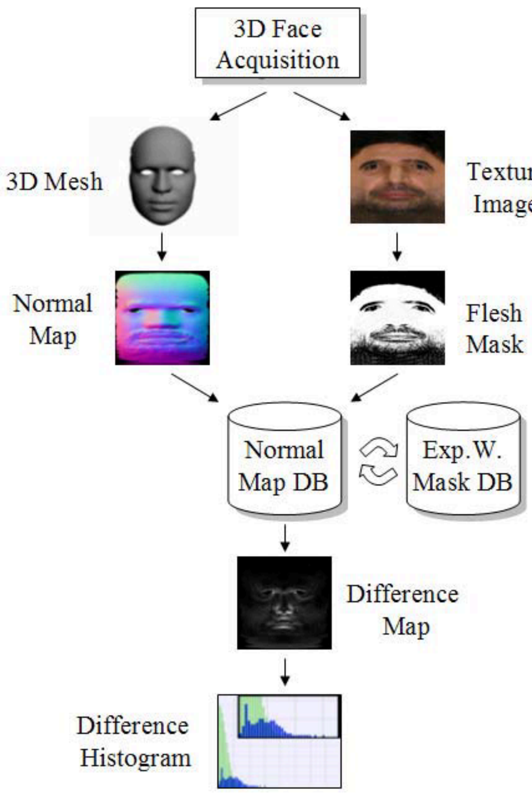


Figure 1 – Method Workflow

A. 3D+2D Face Capturing

As the proposed method works on 3D polygonal meshes mapped with color texture, we firstly need to acquire actual faces and to represent them as polygonal surfaces. Laser or structured light scanning technology could be used to this purpose, capturing range and color data and then converting them in 3D vertices and textured polygons. Stereoscopic imaging is another feasible option for 3D face digitizing which relies on 2D images shot by known angles (a precise calibration is required) and reconstructs a 3D mesh conform to a set of feature points previously correlated between the different views. Even if any of these technologies could provide valid 3D data for the presented method, we opted for a feature based mesh warping technique based on the work by Lee and Magnenat-Thalmann [15] because it requires a simple equipment without critical calibration (a couple of digital cameras shooting from front and side position, see Fig. 2-a) and it is more likely to be adopted in a real application.

In fact, though this image based 3D mesh reconstruction method delivers an inferior face shape accuracy compared to real 3D scanning, it proved to be sufficient for recognition, offering additional advantages, such as precise mesh alignment in 3D space thanks to the warp based approach, facial texture generation from the two captured orthogonal views and its automatic mapping onto the reconstructed face geometry. A more detailed description of this technique and further detail on the enrolment procedure adopted during the experiments of this study are given in section 3.

B. Sampling Mesh Normals

At this point we can store normals of mesh M in a bidimensional matrix N with dimension $l \times m$, to represent face geometry. To this purpose we have to sample the previously mapped geometry and quantize the length of the three scalar components of each normal. Therefore we assign to each pixel (i, j) in N, with $0 \le i < l$ and $0 \le j < m$, the three scalar components of the normal to the point of the mesh surface with mapping coordinates $(l/i,\ m/j)$. The resulting sampling resolution is $1/l$ for the s range and $1/m$ for the t range. The normal components are stored in pixel (i, j) as RGB color components. We refer to the resulting matrix N as the normal map of mesh M. A normal map with a standard color depth of 24 bit allows 8 bit quantization for each normal component, this precision proved to be adequate for the recognition process (Fig 2-c).

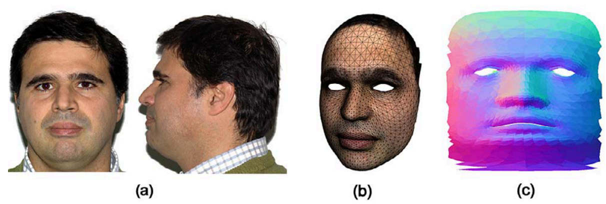


Figure 2 – Captured Face Images (a), 3D Face Mesh (b), Normal Map (c)

C. Basic Normal Map Comparison

When the sampling phase is completed, we can register the new face, i.e. its normal map, in the reference database, or perform a search through it to find a matching subject.

To compare any two face meshes M_A and M_B (namely their normal maps N_A and N_B) we compute the angle included between each pairs of normals represented by colors of pixels with corresponding mapping coordinates, and store it in a new map D. Geometrically, the angle θ from two vector v and w is given by:

$$\arccos(v \times w) \quad (1)$$

as v and w are normalized then $\|v\| \times \|w\| = 1$.

Now, as each pixel (x_{N_A}, y_{N_A}) in N_A has corresponding color components $(r_{N_A}, g_{N_A}, b_{N_A})$ and each pixel (x_{N_B}, y_{N_B}) in N_B has corresponding components $(r_{N_B}, g_{N_B}, b_{N_B})$ the angle included between the normals represented by each pair of pixel with $x_{N_A} = x_{N_B}$ and $y_{N_A} = y_{N_B}$ is given by:

$$\theta = \arccos\left(r_{N_A} \cdot r_{N_B} + g_{N_A} \cdot g_{N_B} + b_{N_A} \cdot b_{N_B}\right) \quad (2)$$

with components opportunely normalized from color domain to spatial domain, so $0 \le r_{N_A}, g_{N_A}, b_{N_A} \le 1$ and $0 \le r_{N_B}, g_{N_B}, b_{N_B} \le 1$. The angle $0 \le \theta < \pi$ is stored in a bidimensional $m \times n$ matrix D which is a gray-scale image (see Fig. 3).

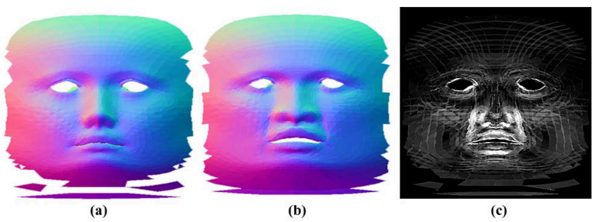


Figure 3 – Comparison of two Normal Maps (a) and (b) and the resulting Difference Map (c)

To reduce the effects of residual face misalignment during acquisition and sampling phases, we calculate the angle θ using a $k \times k$ (usually 3×3 or 5×5) matrix of neighbour pixels.

Summing every gray level in D results in histogram $H(x)$ that represent the angular distance distribution between mesh M_A and M_B as shown in Fig 4. On the X axis we represent the resulting angles between each pair of comparisons (sorted from 0° degree to 180° degree), while on the Y axis we represent the total number of differences found.

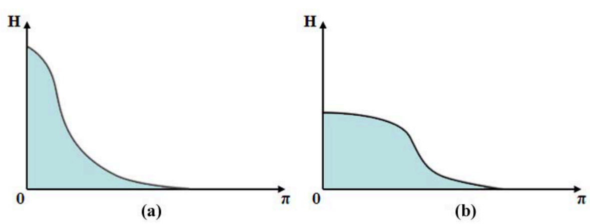


Figure 4 –Angular Distance Distribution in one-to-one comparison: (a) similar faces, (b) not similar faces

This means that two similar faces will have an histogram $H(x)$ with very high values on little angles, while two distinct faces will have differences more distributed. We define a similarity score through a weighted sum between H and a Gaussian function G, as in:

$$similarity\_score = \sum_{x=0}^{k} \left(H(x) \cdot \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} \right) \quad (3)$$

Varying σ and k is possible to change recognition sensitivity .

D. Beard Masking

The presence of beard with variable length covering a portion of the face surface in a subject previously enrolled without it (or vice-versa), could lead to a measurable difference in the overall or local 3D shape of the face mesh (see Fig. 5). In this case the recognition accuracy could be affected resulting, for instance, in a higher False Rejection Rate *FRR*. To improve the robustness to this kind of variable facial features we rely on color data from the captured face texture to mask the non-skin region, eventually disregarding them during the comparison.

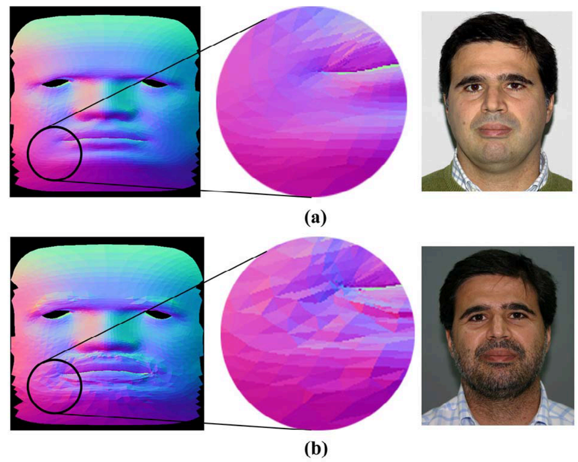


Figure 5 –Normal Maps of the same subject enrolled in two different session with and without beard

We exploit flesh hue characterization in the HSB color space to discriminate between skin and beard/moustaches/eyebrows. Indeed, the hue component of each given texel is much less affected from lighting conditions during capturing then its corresponding RGB value. Nevertheless there could be a wide range of hue values within each skin region due to factors like facial morphology, skin conditions and pathologies, race, etc., so we need to define this range on a case by case basis to obtain a valid mask. To this aim we use a set of specific hue sampling spots located over the face texture at absolute coordinates, selected to be representative of the full flesh tonal range and possibly distant enough from eyes, lips and typical beard and hair covered regions. This is possible because each face mesh and its texture are centered and normalized during the image based reconstruction process (i.e. the face's median axis is always centered on the origin of 3D space with horizontal mapping coordinates equal to 0.5), otherwise normal map comparison would not be possible. The sampling spots locations were selected empirically (see Fig. 6-a), evaluating the union of all the non skin regions from all the subject available in the face database and disregarding it.

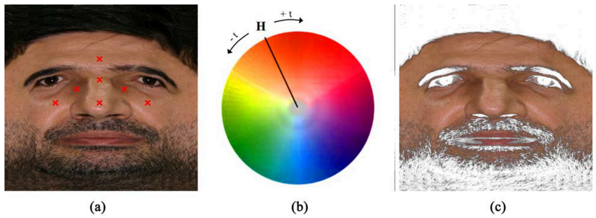


Figure 6 – Flesh Hue sampling points (a), Flesh Hue Range (b) non-skin regions highlighted in white (c)

We could use a 2D or 3D technique to locate main facial features (eye, nose and lips) and to position the sampling spots relative to this features, but even these approaches are not safe under all conditions.

For each sampling spot we sample not just that texel but a 5x5 matrix of neighbour texels, averaging them to minimize the effect of local image noise. As any sampling spot could casually pick wrong values due to local skin color anomalies such as moles, scars or even for improper positioning, we calculate the median of all resulting hue values from all sampling spots, resulting in a main Flesh Hue Value *FHV* which is the center of the valid flesh hue range. We therefore consider belonging to skin region all the texels whose hue value is within the range: $-t \leq FHV \leq t$, where t is a hue tolerance which we experimentally found could be set below 10° (see Fig. 6-b). After the skin region has been selected, it is filled with pure white while the remaining regions are filled with pure black. The resulting 1 bit flesh mask (see Fig. 7) multiplied by the difference map during comparison cuts non-skin regions which could erroneously affect the recognition.

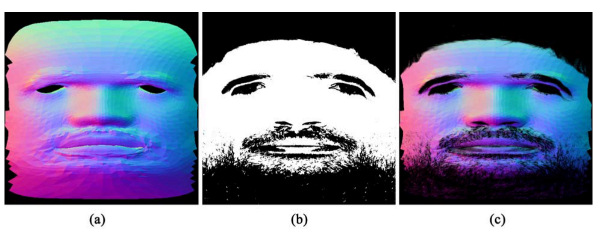


Figure 7 –Normal Map in a subject with beard (a), Flesh Mask (b), Normal Map - Flesh Mask (c)

E. Addressing Facial Expressions

To improve robustness to facial expressions we introduce the expression weighting mask, a subject specific pre-calculated mask aimed to assign different relevance to different face regions.

This mask, which shares the same size of normal map and difference map, contains for each pixel an 8 bit weight encoding the local rigidity of the face surface based on the analysis of a pre-built set of nine facial expressions of the same subject (see section 3 for details).

Indeed, for each subject enrolled, each of ten expression variations is compared to the neutral face resulting in ten difference maps. The average of this set of difference maps specific to the same individual represent its expression weighting mask.

More precisely, given a generic face with its normal map N_0 (neutral face) and the set of normal maps N_1, N_2, …, N_n (the expression variations), we first calculate the set of difference map D_1, D_2, …, D_n resulting from {`$N_0$ - $N_1$', `N_0 - N_2', …, `$N_0 – N_n$'}. The average of set {D_1, D_2, …, D_n} is the expression weighting mask which is multiplied by the difference map in each comparison between two faces. The expression weighting mask allows to better cope with bending facial regions.

III. Experiments and Discussion

As one of the aims in experiments was to test the performance of the proposed method in a realistic operative environment, we decided to build a 3D face database based on 2D color pictures instead of range data. More precisely, every face model in the database has been created deforming a pre-aligned prototype polygonal face mesh to closely fit a set of facial features extracted from front and side images of each individual enrolled in the system. The face capture station featured two digital cameras with external electronic strobes shooting simultaneously with a shutter speed of 1/250 sec. while the subject was looking at a blinking led to reduce posing issues. Face geometry and color texture are reconstructed following the mesh warping technique described in [15]. Indeed, for each enrolled subject a set of corresponding facial features extracted by a structured snake method from the two orthogonal views are correlated first and then used to guide the prototype mesh warping performed through a Dirichlet Free Form Deformation. The two captured face images are aligned, combined and blended resulting in a color texture precisely fitting the reconstructed face mesh through the feature points previously extracted. The prototype face mesh used in the dataset has about 7K triangular facets, and even if it is possible to use mesh with higher level of detail we found this resolution to be adequate for face recognition. This is mainly due to the optimized tessellation which privileges key area such as eyes, nose and lips whereas a typical mesh produced by 3D scanner features almost evenly spaced vertices. Another remarkable advantage involved in the warp based mesh generation is the ability to reproduce a broad range of face variations through a rig based deformation system. This technique is commonly used in computer graphics for facial animation [16, 17] and is easily applied to the prototype mesh linking the rig system to specific subsets of vertices on the face surface. Any facial expression could be mimicked opportunely combining the effect of the rig controlling lips, mouth shape, eye closing or opening, nose tip or bridge, cheek shape, eyebrows shape, etc. The facial deformation model we used is based on [16] and the resulting expressions are anatomically correct. We augmented the 3D dataset of each enrolled subject through the synthesis of nine additional expressions leading to the related weighting mask: neutral, rage (moderate), fear, smile (closed), doubt, surprise (moderate), rage (extreme), closed eyes, surprise (extreme), disgust (see Fig. 8 below).

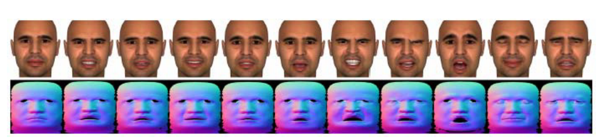


Figure 8 – Synthetically generated expression variations and corresponding normal maps

We chosen these particular expressions trying to exaggerate the range of expression variations usually considered in most studies. The full database used during experiments includes 135 different persons (87 males and 48 females, age ranging from 19 to 65) each one with ten synthesized expression variations. To assess the effectiveness of the flesh mask to improve recognition in presence of beard, 35 among male subject in database where enrolled a second time featuring beard of various length, color and shape after a period of time variable from two weeks to two months since the first enrolment session.

We performed three different kinds of experiments using normal maps and color textures sized 128×128pixels which proved to be adequate for the purpose, while the Gaussian function $G(x)$ has been built choosing σ=4.5 and k=50. The first test was meant to measure the recognition rate through a one-to-many comparison performed on a probe set of faces with various expressions against a gallery set of neutral faces, and it reached a rate of 100%.

Since the lack of a reference facial database, such as the FERET [18] is for 2D face recognition, any direct comparison between various 3DFR approaches on different dataset is not conclusive. Nevertheless, in Table 1 we resume the performance of the methods cited in section I and the correspondent database used. Only a small subset of 3DFR techniques address expressive variations while the proposed method is the only one to specifically address the presence/absence of beard.

In following experiment we tested the method's Precision/Recall with and without the expression weighting mask, with and without the flesh mask and with both of them combined respectively (see Fig. 9 below).

METHOD	Database			Express. Tolerant	Beard Tolerant	Recogn. Rate %
	Size	No. Subjects	Variations			
Gordon [2]	24	8	2 facial expressions	NO	NO	100%
Achermann et al. [3]	240	24	10 poses	NO	NO	100%
Achermann et al. [4]	240	24	10 poses	NO	NO	100%
Tanaka et al. [5]	37	37	-	NO	NO	100%
Hesher et al. [6]	185	37	5 facial expressions	NO	NO	90%
Medioni et al. [7]	700	100	7 poses	NO	NO	100%
Bronstein et al. [8]	-	157	-	YES	NO	-
Chang et al. [9]	-	275	-	NO	NO	98.8%
Tsalakanidou et al. [10]	80	40	-	NO	NO	99%
Papatheodorou et al. [11]	~900	62	5 poses and 3 facial expressions	NO	NO	-
Beumier et al. [12]	~720	120	3 poses for 2 sessions	NO	NO	98%
Wang et al. [13]	300	50	-	YES	NO	> 90%
Proposed Method	1485	135	11 facial expressions	YES	YES	100%

Table 1. Some 3D face recognition methods resumed.

The results showed in Fig. 9-a were achieved comparing in one to many modality a query set with one expressive variations to an answer set composed by one neutral face plus ten expression variations and one face with beard.

In Fig. 9-b are shown the results of one to many comparison between subject with beard and an answer set composed of one neutral face and ten expressive variations. Finally for the test reported in Fig. 9-c the query was an expression variation or a face with beard, while the answer set could contain a neutral face plus ten associated expressive variations or a face with beard. The three charts clearly show the benefits involved with the use of both expressive and flesh mask, specially combined together.

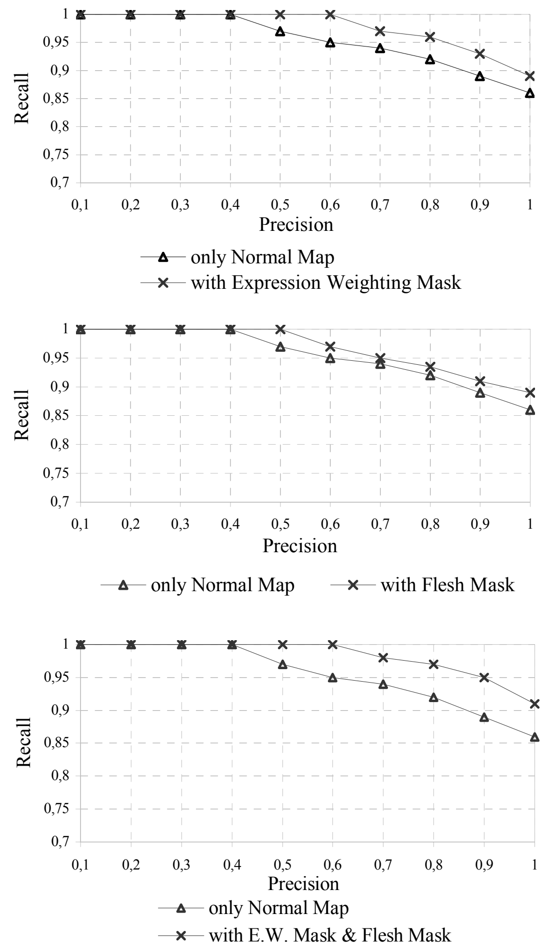


Figure 9 – Precision/Recall Testing

The purpose of the third experiment, whose results are showed in Fig.10, is to specifically measure the Cumulative Match Score improvement due to the implementation of the flesh mask, comparing each of the 35 face with beard to the neutral faces without beard. This test highlights the advantage of flesh mask in term of recognition accuracy, despite the involved reduction of useful surface in masked normal maps.

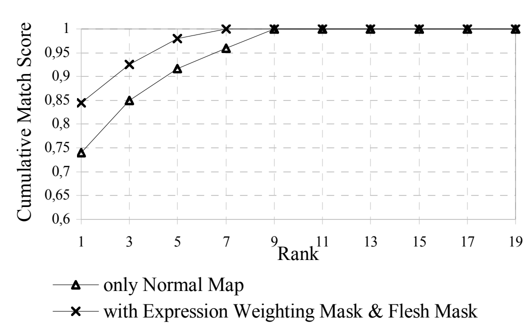


Figure 10 – Cumulative Match Score

The proposed recognition algorithm has a low computational cost (as shown in section 2) with a $O(n)$ time complexity (where n is the total number of pixels in the normal map) and it required approx. 4 milliseconds on a P4/3.4 Ghz based PC for a single one to one comparison, a figure allowing to use the method in a one to many application even on a large gallery.

IV. CONCLUSIONS AND FUTURE RESEARCH

We presented a 3D face recognition method based on 3D geometry and color texture, aimed to improve robustness to presence/absence of beard and to expressive variations. It proved to be simple and fast and experiments conducted showed high average recognition rate and a measurable effectiveness of both flesh mask and expression weighting mask. Ongoing research will implement a true multi-modal version of the basic algorithm with a second recognition engine dedicated to the color info (texture) which could further enhance the discriminating power. On the other side we are in the process to test the method on a new 3D face dataset built with a structured light scanner from Inspeck Corp. to assess the recognition accuracy on more disorganized and noisy, but potentially much more precise 3D data. As the foundation of the presented approach is a 2D bitmap representing geometry and color info and simple processing at pixel level, we are also investigating the possibility to implement the recognition engine on Graphics Processing Units (GPUs). Exploiting the high parallelism of this specialized programmable processors, could bring a big performance leap, allowing to operate even on huge database for effective large scale one-to-many recognition applications.

REFERENCES

[1] K.W. Bowyer, K. Chang, P. A. Flynn, Survey of 3D and Multi-Modal 3D+2D Face Recognition, ICPR 2004.

[2] G. Gordon, Face recognition based on depth and curvature features. Computer Vision and Pattern Recognition (CVPR), pages 108-110, June 1992.

[3] B. Achermann, X. Jiang, H. Bunke, Face recognition using range images. International Conference on Virtual Systems and MultiMedia, pages 129ñ 136, 1997.

[4] B. Achermann, H. Bunke, Classifying range images of human faces with Hausdorff distance, 15-th International Conference on Pattern Recognition, pages 809-813, September 2000.

[5] H. T. Tanaka, M. Ikeda, H. Chiaki, Curvature-based face surface recognition using spherical correlation principal directions for curved object recognition, Third International Conference on Automated Face and Gesture Recognition, pages 372-377, 1998.

[6] C. Hesher, A. Srivastava, G. Erlebacher, A novel technique for face recognition using range images, Seventh Int'l Symposium on Signal Processing and Its Applications, 2003.

[7] G. Medioni, R. Waupotitsch, Face recognition and modeling in 3D. IEEE International Workshop on Analysis and Modeling of Faces and Gestures (AMFG 2003), pages 232-233, October 2003.

[8] A. M. Bronstein, M. M. Bronstein, R. Kimmel, Expression-invariant 3D face recognition, AudioandVideo-Based Person Authentication (AVBPA 2003), LCNS 2688, J. Kittler and M.S. Nixon, eds.:62-70,2003.

[9] K. Chang, K. Bowyer, P. Flynn, Face Recognition Using 2D and 3D Facial Data, in Proceedings of the ACM Workshop on Multimodal User Authentication, pp. 25-32, December 2003.

[10] F. Tsalakanidou, D. Tzovaras, M. G. Strintzis, ”Use of depth and color eigenfaces for face recognition” , in Pattern Recognition Letters, vol. 24, No. 9-10, pp. 1427-1435, Jan-2003.

[11] T. Papatheodorou, D. Rueckert, Evaluation of Automatic 4D Face Recognition Using Surface and Texture Registration, in Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 321-326, May 2004, Seoul, Korea.

[12] C. Beumier and M. Acheroy, Automatic Face verification from 3D and grey level cues, 11th Portuguese Conference on Pattern Recognition (RECPAD 2000), May 2000, Porto, Portugal.

[13] Y. Wang, C. Chua, and Y. Ho, Facial feature detection and face recognition from 2D and 3D images, Pattern Recognition Letters, 23:1191-1202, 2002.

[14] X. Gu, S. Gortler, H. Hoppe, Geometry image, ACM SIGGRAPH 2002, pages 355-361.

[15] W. Lee, N. Magnenat-Thalmann. Fast Head Modeling for Animation., Journal Image and Vision Computing, Number 4, Elsevier, pp. 18: 355-364, 2000.

[16] Y. Lee, D. Terzopoulos, and K. Waters, Realistic modeling for facial animation, in Proc. SIGGRAPH 95, Los Angeles, CA, Aug. 1995, pp. 55--62.

[17] V. Blanz and T. Vetter, A morphable model for the synthesis of 3D faces, in Proc. SIGGRAPH 99, Los Angeles, CA, Aug. 1999, pp. 187--194.

[18] J. P. Phillips, H. Moon, A. S. Rizvi, P. J. Rauss, ”The FERET Evaluation Methodology for Face-Recognition Algorithms,” in IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 22, no. 10, pp. 1090-1104, Oct. 2000.

PLAY WITH VIDEO: AN INTEGRATION OF HUMAN MOTION TRACKING AND INTERACTIVE VIDEO

Timothy K. Shih, Hui-Huang Hsu, Chia-Ton Tan and Louis H. Lin
Multimedia Information Network Lab
Department of Computer Science and Information Engineering, Tamkang University, Taiwan
E-mail: tshih@cs.tku.edu.tw

ABSTRACT

Human motion tracking mechanisms need to track multiple points on a human skeleton. Whether it is based on video or sensors, it is computationally expensive if the tracked skeleton needs to be embedded into a scenario video. We propose a systematic strategy for human action tracking, with an important extension to integrate tracked skeleton with an interactive video player. The interactive video player can detect and understand a restricted set of human motions and react by jumping to particular video segments pre-recorded. As a consequence, the system allows a player to fight with an avatar or a real person in an interactive video game.

Keywords: Human Motion Tracking, Interactive Video, video game, human computer interaction

1. INTRODUCTION

Human motion tracking was developed in the past few years. The MARG sensors are used to develop a system which can embed skeleton into a virtual environment [1]. Instead of using sensors, video-based tracking strategy was proposed [3, 4]. The difficulty of video-based approach is on how to deal with incomplete motion such as occlusion or bad viewing angle. Another difficulty is due to variations of light sources. Since sensors are expensive in general, video-based approach is usually used in applications. In order to embed avatars into a virtual reality environment, 3D coordinate reconstruction methods need to be used [2, 5, 6].

We have implemented a system which takes a further step to integrate human actions with an interactive video player. Figure 1 shows our system setup. The player wears a black suit, with 16 track points (TPs) placed based on human skeleton. Two video cameras are located in front and on the side of the player to extract 2 sets of 2D coordinate for 3D coordinate reconstruction. The 16 track points are identified and mapped to the skeleton for motion understanding. Finally, the skeleton is mapped onto the interactive video, which contains special trigger points to detect human motions and react on the motions. Application of our system can be used in interactive video games, simulations, training, and others.

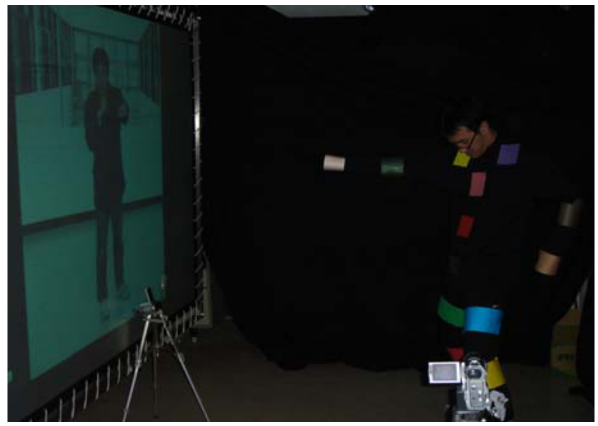


Figure 1: Play with Video

2. TRACKING AND INTERACTION

As illustrated in the flowchart in figure 2, the tracking and interaction process involves five steps (illustrated in different zones) of computation:

(1) Initialization and video camera synchronization
(2) Background and track points separation
(3) Track points identification
(4) Skeleton reconstruction on background video
(5) Motion understanding and interaction control

These steps will be discussed in the following sections.

2.1. Initialization and video camera synchronization

Background need to be separated from track points in order to identify track points. In the first step (marked as (1) in figure 2), an initial video background can be obtained from a video segment of 30 to 60 frames, where moving objects are removed and holes are inpainted. We use a fast image inpainting algorithm which relies on a diffusion kernel. In addition, since we use two video cameras for 3D coordinate reconstruction, the two cameras needs to be synchronized. Our strategy requires the player to stand steady for at least 2 seconds. A variation threshold for video changes is set to tell the steady situation. Thus, in the first step, the player need to move around for a few seconds, stand steady for 2 seconds, then our system start the second step (marked as (2)) to track the skeleton.

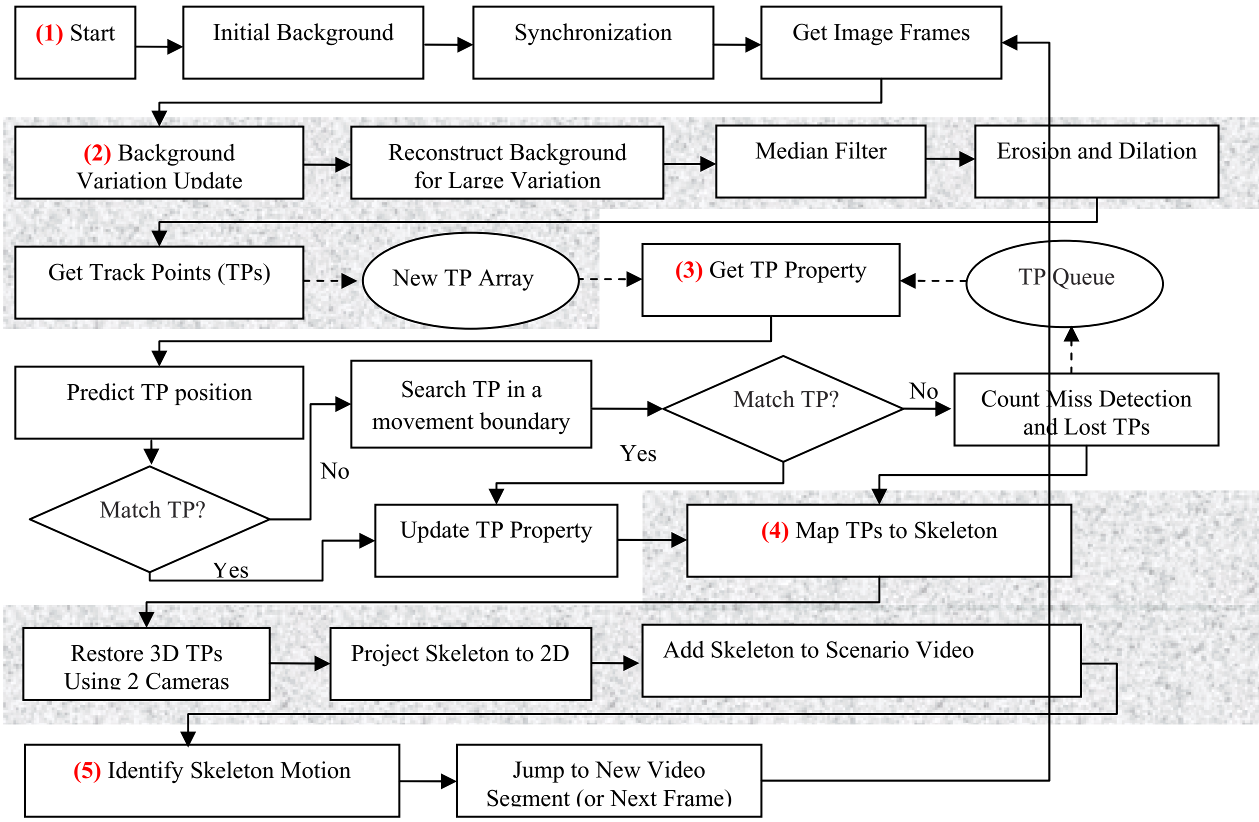


Figure 2: Algorithm of Video Tracking and Interaction

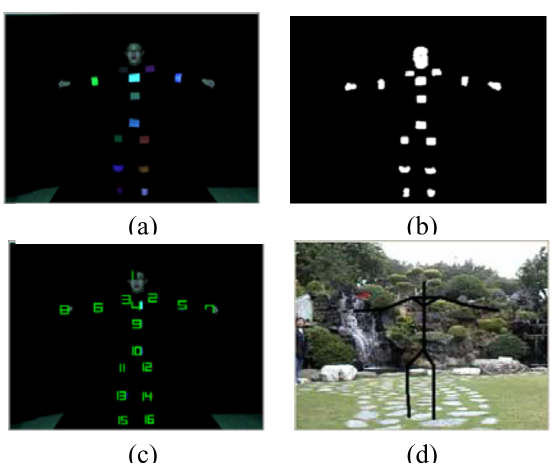


Figure 3: Tracking Process (a) Original Video (b) Track Points (c) Track Points Identification (d) Add Skeleton to Scenario Video

2.2. Background and track points separation

Since light sources to track points are quite difficult to stay stable, it is necessary to update background dynamically in order to obtain a better separation result of background and track points. Assuming that $B_k(x, y)$ and $B_{k+1}(x, y)$ are values of point (x, y) on the k_{th} and the $k+1_{th}$ frames, respectively. And, $I_k(x, y)$ is the value of point (x, y) on the coming k_{th} frame. We use

$$B_{k+1}(x,y)=\begin{cases} B_k(x,y)+\alpha\left[B_k(x,y)-I_k(x,y)\right]\ ,\ \alpha=1 \\ \quad \text{if } I_k(x,y)\in \text{object point} \\ B_k(x,y)+\beta\left[B_k(x,y)-I_k(x,y)\right]\ ,\ \beta=0.01 \\ \quad \text{if } I_k(x,y)\notin \text{object point} \end{cases}$$

This strategy works if small variation of light source occurs on background. However, if the variation is too large, the background should be reconstructed. After that, we use median filter to exclude isolated points. Also, we use two morphology operators (erosion and dilation) to perform the close operation. The close operation is able to eliminate small blocks. Figure 3 shows the original video in 3a and the track points in 3b after the second step.

2.3. Track points identification

The most challenge issue of multiple object tracking is to identify each and all objects. We maintain a track point array to store properties (location and color information) of all track points. In step (3), we use seed filling algorithm on the original video for basic color segmentation. Boundaries of track points are roughly computed, to obtain the center of each track point. Each record in the track point array contains a location of the center and the HSI color information. To precisely track each point, two issues need to be solved. Track points moves in a direction which is hard to predict; and, track points can be cloaked (occlusion). To solve the first problem, prediction and searching mechanisms are used. The second problem can be partially solved with multiple cameras. However, it is possible to solve the second problem using a unique color for each track point and searching in a range of movement. And, this is our approach. A sample predication mechanism which computes the average movement vector of each track point in the latest 30 frames is used. If the prediction of track point matches (within a small threshold of center location), the corresponding track point is updated in the array. Otherwise, the system searches the track points in a small boundary (eight directions to speed up computation). Track point properties are updated if there is a match. Otherwise, we count the track point as miss detection. An evaluation of miss detection is given in section 3. When a miss detection occurs, we keep the miss detected track point in a track point queue. In the next iteration, track points are searched against the queue with a higher priority. The search will focus on the color of track points. Our preliminary experience shows that, with the setup of our environment, the miss detection rate is tolerable. In figure 3c, track points are identified with numbers.

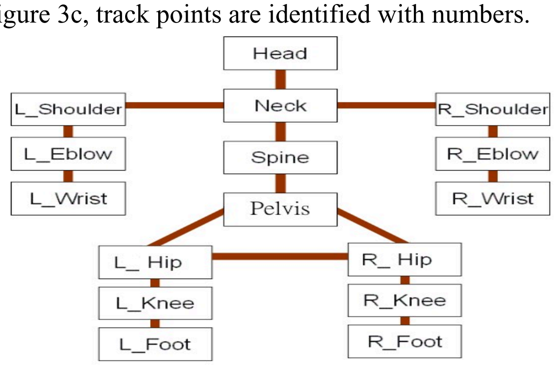


Figure 4: Skeleton with 16 Track Points

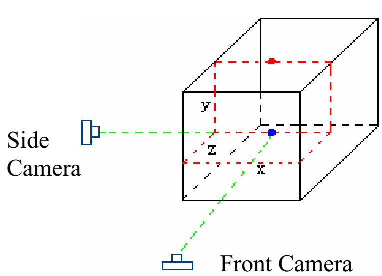


Figure 5: Coordinates and Front and Side Cameras

2.4. Skeleton reconstruction on background video

After the track points are identified, we need to map them to a skeleton which represents a human body. The mapping process is performed as soon as all TPs are identified in the first run. Since TPs will be tracked with unique IDs, remapping to skeleton is thus not necessary. The mapping strategy takes a simple heuristic rule: assuming that the play is not up side down or doing a handspring (i.e., flip). The mapping strategy follows:

1. Compute the fulcrum of 16 track points based on the initial posture (see figure 3), set fulcrum to be the Pelvis (see figure 4)
2. Follow the vertical line up to find Spine, Neck and Head. A horizontal threshold of a few pixels is used in case that the 4 track points of human body are not aligned vertically.
3. Split the rest 12 TPs in the skeleton to left and right, according to the 4 track points mapped in the above step. Following the Pelvis to find L_Hip, L_Knee, and L_Foot and following the Neck to find L_Shoulder, L_Eblow, and L_Wrist. Spatial relations of TPs are used as the heuristic.
4. Do the same for right hand side track points.

Only the track points captured by the front camera are used to identify the track points. However, to reconstruct 3D coordinates for all track points, it is necessary to use the side camera (along the x-axis to the left of the box in figure 5). To restore 3D coordinates, we use the following strategy:

1. For each track points on both cameras, find the differences of coordinates on y-axis.
2. Minimizing the differences to align TPs on y-axis using dynamic programming.
3. Take the x-coordinates from the front camera and the z-coordinates from the side camera.

After the 3D coordinates are computed, it is necessary to perform a projection of these coordinates to 2D space. One may argue that it is not necessary for 3D coordinate reconstruction and projection, and using 2D coordinates will be enough. However, to make the interaction realistic, 3D information is necessary. For instance, the player must fell that a punch is indeed located on an object in the video by moving forward his arm. In addition, if virtual reality avatar is used, 3D coordinate is required. After the projected 2D coordinates are obtained, we map the skeleton onto a scenario video. The scenario video follows MPEG-2, with an important extension allows a hyper jump among video segments. Hyper jump tags are embedded in the user defined data section of standard MPEG-2 video clips. The scenario video can be a pre-recorded video game or a training video for customers. For performance consideration, only a small section of video is stored in memory to speed up accessing time. Figure 3d shows the skeleton on a scenario video.

We use OpenGL [7][8] technology to reconstruct the 3D environment. OpenGL provides high performance with 3D graphics rendering. With the 3D graphics projection, we can get the all direction of projection to the 2D video frame.

To avoid the heavy loading of capturing, object tracking, 3D projection and rendering, we separate these processes to two computers with network connection. The client-side processes the capturing and object-tracking. After these heavy steps, the client-side will generate the tracking points, and transfer these point coordinate to the server-side. The server-side receives these data, generates the 3D scene. And the server-side can also do off-screen rendering the 3D scene to another 2D video frame.

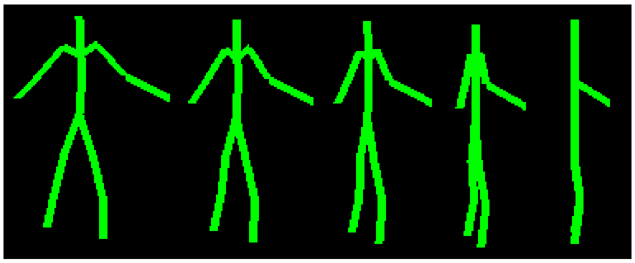


Figure 6: 3D Skeleton Representation on Server-Side

2.5. Motion understanding and interaction control

The last step is to control interaction. For our prototype system, we use a simple strategy to store skeleton motions. Each of the track points, L_Wrist, R_Wrist, L_Foot, and R_Foot, is recorded in the track point array with a motion vector, in addition to TP coordinate and color information. A motion vector can be one of the eight directions (i.e., East, West, South, North, Southeast, Southwest, Northeast, and Northwest) and the directions are extended to include z-movements. The motion vectors of the 4 track points are analyzed by an interactive controller. With special designed 'hot spot' in the interactive video, the hot spot is able to tell which of the 4 track points is engaged with a certain motion vector. Thus, a video hyper jump can be performed. Figure 1 shows our system with an interactive video made by real person. Note that, each section of the video has a small movement of the avatar to response to a specific trigger from a track point and its motion vector. Our preliminary system contains only hard coded video segments. An authoring tool is underdevelopment to enable other scenario video clips.

3. EXPERIMENT RESULTS AND ANALYSIS

We evaluate the experiment system from two perspectives: the precision rate of tracking and the performance of the system. High tracking precision avoids misunderstanding of motion hence the system reacts as the player expected. Human motion tracking is affected by three important issues: changing light sources, speed of motion, and track point occlusion. We tested 12 video clips to analyze rate of lost track points and rate of miss detection. They are summarized in table 1.

Table 1: Rates of Tract Point Detection

Video	No of Frames	Lost TP Rate	Miss Detection Rate
1	600	0.8 %	1.1 %
2	403	4.5 %	7.2 %
3	350	3.9 %	6.7 %
4	545	1.2 %	1.7 %
5	570	1.2 %	0.2 %
6	330	1.3 %	1.0 %
7	420	2.7 %	3.0 %
8	391	0.3 %	0.1 %
9	818	0.3 %	0.8 %
10	514	0.36 %	1.3 %
11	467	3.7 %	4.3 %
12	687	1.0 %	1.8 %
Average		1.77 %	2.43 %

Video 2 and 3 have a relatively unstable light source and higher occlusion. The miss detection rates are 7.2 and 6.7 percents. Video 7 and 11 have a stable light source. However, track points move faster. The miss detection rates are 3.0 and 4.3 percents. Video 8 and 9 have very low rates since light source is stable. Intuitively speaking, miss detection rates get higher due to changing light source the most. Occlusion and motion speed are secondary factors. Our average miss detection rate is 2.43%, which is tolerable.

The performance of our system is not as we have expected but is reasonable. We use 320 by 240 as the resolution of the two video cameras. The resolution of scenario video is also 320 by 240. We try to lower the resolution of the two cameras. But, precision of tracking is decreased. Step three of the algorithm (i.e., track point identification) is computational expensive, especially in searching miss matched track points. In step four (i.e, skeleton reconstruction on background video), restoration of 3D information takes time. However, we use a simplified approach to use the front camera for skeleton mapping. But, the side camera is used only in the identification of skeleton motion. Thus, project from 3D to 2D is simplified to speed up computation. The system is running on a Pentium 4 2.8GHz computer with 1.5 GB RAM. Preliminary experience shows that, player is able to interact with the video with limited motions to control video reactions.

4. CONCLUSIONS

It is interesting to combine video tracking technique with interactive video to develop a system for video-based games. Our approach involves five steps of schemes to realize the system on a Window based environment. Our experience shows that the approach is feasible on ordinary personal computers, instead of using special hardware devices such as Play Station 2. Our future work includes the design and implementation of an authoring system for the scenario video and motion identification. We hope software on PCs will allow players to design their own video games using PCs and ordinary cameras.

REFERENCES

[1] Eric R. Bachmann, Robert B. McGhee, Xiaoping Yun, and Michael J. Zyda "Inertial and magnetic posture tracking for inserting humans into networked virtual environments," Proceedings of the ACM symposium on Virtual reality software and technology, November 2001, pp. 9-16.

[2] D.E. Difranco, T.J. Cham, and J.M. Rehg. "Recovery of 3-D figure motion from 2-D correspondences," in CVPR, 2001, pp. I307-I314.

[3] Z. Luo, Y. Zhuang, F. Liu, and Y. Pan, "Incomplete Motion Feature Tracking Algorithm in Video Sequences," in Proceedings of the IEEE 2002 International Conference on Image Processing, New York, USA, 2002, pp. III/617-III/620.

[4] L. Sigal, S. Bhatia, S. Roth, M.J. Black, and M. Isard. "Tracking loose-limbed people," in CVPR, 2004. pp. I421-I428.

[5] C. Sminchisescu and B. Triggs. "Kinematic jump processes for monocular 3D human tracking," In CVPR' 03. pp. I/69-I/76.

[6] KangKang Yin, and Dinesh K. Pai "Intuitive interfaces for animation: FootSee: an interactive animation system," in Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer animation, July 2003, pp. 329-338.

[7] OpenGL, http://www.opengl.org/

[8] SGI, http://www.sgi.com/products/software/opengl/

A Design for Speaker Determination in Video Conferencing: An Application of Speaker Recognition

L. Bateman
Digital Media Laboratory
University of Calgary
2500 University Dr. N.W.
Calgary, Alberta, Canada T2N 1N4
1 403 220 4315

J. R. Parker
Digital Media Laboratory
Department of Computer Science
University of Calgary
1 403 220 6784
parker@cpsc.ucalgary.ca

ABSTRACT

There are many reasons why a video conferencing system should keep track of who is speaking. Knowing who is speaking would be an essential part of a creating a transcription, and the system can use speaker identity to control the camera appropriately. Speaker determination can be done by using special hardware that triangulates sound to a source in three-dimensional space, but most existing computers do not have such equipment. We propose to identify speakers by recognizing their voices.

Keywords

sound, speaker recognition

INTRODUCTION

In the context of a video conference or an online classroom situation, it is often useful to know the identity of the person speaking. The information can be used to control video cameras, during transcription, and even to just assign credit and keep track of participation. Special purpose hardware can be constructed to triangulate the source of a sound in a room, and this could be use to determine the source of speech. The equipment is not 'off the shelf', though, and is uncommon. It would not help at all in situations where the conference was recorded and played back later.

We propose the use of audio speaker identification techniques to identify speakers. These methods are implemented in software, and identify an individual from many by characterizing basic parameters of their speech signals, recorded as digital audio.

SPEAKER IDENTIFICATION

Speaker recognition is not the same as speech recognition[2,3,12,14]. In speech recognition the goal it to determine what is being said; in speaker recognition, the goal is to determine who is saying it. Identification is performed my making measurements, also known as features, of the speech sample and comparing those against the measurements of the same features made of known speakers. If a match to within a specified tolerance is made, then the speaker is identified, otherwise the speech sample is said to be *rejected*. There are some traditional ways to approach speaker recognition, and standard data sets to use.

Linear Prediction (LP)

Linear prediction (LP) is an autoregressive method of feature extraction [8,9]. This means that what the speech is doing at time t is determined by what the speech was doing at times t-N to t-1. In other words, the previous behavior of the speech is used to predict the future behavior of the speech. It has the effect of smoothing the spectral envelope of the speech. In other words, the complex local fluctuations of the input speech waveform are somewhat smoothed away. This is somewhat desirable because some of these fluctuations are probably due to noise.

Linear Prediction is one of the most powerful and important methods of deconvolution and parameterization of the source and filter. The source, or excitation, is the driving force of speech production. It is the air that is pushed from the lungs through the vocal tract. The filter is the vocal tract itself. As the air is pushed through, it changes the air pressure that is created by the force of the air. By the time the air is expelled from the lips, the air pressure creates waves of sound that reflect the vocal tract changes that took place. LPC is explained in detail in [6] and [7]. It is one of the most commonly used features in the literature. It is often either used directly or is the basis of further feature extraction. In fact, the autoregressive idea is used in a slightly different way in many psychoacoustic features, which are discussed later on in this chapter.

Linear prediction is based on the idea that the vocal tract can be modeled by a series of nonuniform, piece-wise acoustic tubes that are joined together.

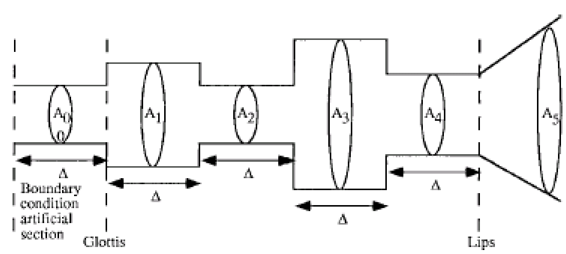


Figure 1: Acoustic Tube Model

Fig. 1 shows a crude acoustic tube model that is used to model the vocal tract. Adjacent sections of the tube vary in shape and diameter. The excitation signal, or source, is the driving force of speech production. It passes through the vocal tract, or filter, from left to right, to produce a speech sound s(n). Most speech models, including linear prediction, attempt to decouple the source from the filter as a preliminary step.

The all-pole LP models a signal s(n) by a linear combination of its past values and a scaled present input. In the time domain, it is as follows:

$$s(n) = -\sum (a_k * s(n-k) + G * (u_n)$$

where s(n) is the present output, p is the prediction order, a_k are the prediction coefficients, s(n-k) are the past outputs, G is a gain scaling factor, and u_n is the present input, which corresponds to the human vocal tract excitation. This is simplified further because in speech applications, only the vocal tract, or filter is kept, and therefore the input u_n is generally removed since it is unknown. Thus we get the following equation, which now depends only on past outputs:

$$\hat{s}(n) = -\sum (a_k * s(n-k))$$

Now the problem of estimating a_k is easier because the source and the filter have been decoupled. The source, u_n, is not modeled by these prediction coefficients, and thus, by ignoring it, it is probably reasonable to assume that some sort of valuable speaker-dependent information that is present in the excitation signal has been lost. The linear prediction coefficients are typically found using a mean-square estimate. It has been found that minimizing the error signal e(n) in this way produces a flat, or band-limited white magnitude spectrum of the error signal, which can be defined as being the difference between the actual signal s(n) and the estimated reconstruction of the signal using the prediction coefficients, s(n):

$$e(n) = s(n) - \hat{s}(n) = s(n) + \sum (a_k * s(n-k))$$

Using s(n) above, the LP Z-domain transfer function is as follows:

$$H(z) = \frac{G}{1 + \sum (a_k * z^{-k})} = \frac{G}{A(z)}$$

where A(z) is known as the p^{th}-order inverse filter.

One criticism that has been made about LPC is that the underlying assumption about the piecewise acoustic tube is incorrect, or inadequate. For example, the acoustic tube model is static, whereas in reality, the human vocal tract is always changing shape. However, the beauty of LPC is in its simplicity, especially when unknown terms such as u_n can simply be factored out of the equation altogether.

Line Spectrum Pair (LSP) Frequencies

Line Spectrum Pair (LSP) frequencies are an alternative spectral representation of the linear prediction coefficients, and are discussed in [15]. Here the use of LSP frequencies in speech data compression is discussed and some of their main properties are described and proven. An explanation of Line Spectrum Pair frequencies is also given in [5].

Line spectrum pairs are essentially a representation of p^{th}-order coefficients of the inverse Z-domain filter A(z) of the LP all-pole representation as follows:

$$A(z) = \frac{1}{2}[P(z) + Q(z)]$$

Then define:

$$P(z) = A(z) + z^{-(p+1)} * A(z^{-1})$$

$$Q(z) = A(z) - z^{-(p+1)} * A(z^{-1})$$

where P(z) and Q(z) are (p+1)-order symmetric and anti symmetric polynomials whose zeros are mapped onto the unit circle in the Z-domain. The zeros of the polynomial P and the zeros of the polynomial Q are interlaced with one another. The frequencies at which these zeros occur are the LSP frequencies. The zeros of the P polynomial are computed using the discrete cosine transform (DCT), whereas the zeros of the Q polynomial are computed using the discrete sine transform (DST).

The DCT is defined as follows:

$$f_j = \sum_{k=0}^{n-1} x_k \cos(\frac{\pi}{n} j(k + \frac{1}{2}))$$

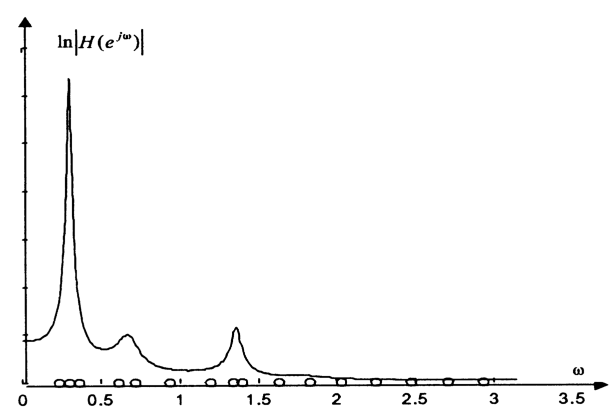
and the DST is defined as:

$$f_j = \sum_{k=0}^{n-1} x_k \sin(\frac{\pi}{n}(j+1)(k + \frac{1}{2}))$$

The DCT and DST are each similar to the DFT, but they both use real numbers and are roughly twice the length of the DFT. The DCT operates on data with even symmetry and is equivalent to the real parts of the DFT. The DST operates on data with odd symmetry and is equivalent to the imaginary parts of the DFT.

LSP frequencies have traditionally been used for speech data compression and quantization. One example of where they are used is in the famous CELP cell phone data compression. The LSP frequencies have been shown to be more stable than their LPC representation, therefore yielding higher accuracies in recognition applications[16].

The following figure illustrates the relationship between the zeros of the P and Q polynomials and the corresponding transfer function H(z). The closer together two adjacent zeros, the more of a peak is seen in the transfer function.



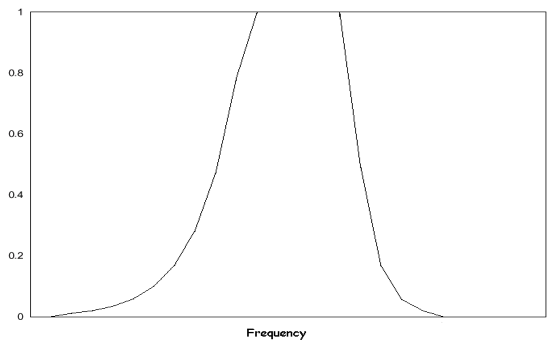


Figure 2: The Shape of Hermansky's Auditory Filters

Perceptual Linear Prediction (PLP)

Hermansky's Perceptual Linear Predictive analysis of speech [4] exploits many of the aforementioned psychoacoustic properties of human hearing and is a popular feature that had been used in many ASR systems. The procedure for analyzing a frame of speech to yield a set of PLP coefficients is described briefly next.

Hermansky begins by applying a hamming window to the time-domain speech signal within the frame to prepare it for frequency-domain analysis. The windowed signal is taken to the frequency domain using Fourier analysis, and the power spectrum is computed:

$$P(w) = \mathrm{Re}\,al[S(\omega)]^2 + \mathrm{Im}\,ag[S(\omega)]^2$$

Next, a filter bank of auditory filters are constructed, simulating what has been found about the basilar membrane's response to sound stimuli. Each auditory filter is quasi-trapezoidal in shape.

The full filter bank is shown in figure 2 above.

First, the critical bands are computed in Bark frequencies, and then the filters themselves are built around them. Given the central frequency of a critical band, the auditory filter is built using:

$$\psi(\Omega) = \begin{cases} 0 & for\ \Omega < -1.3 \\ 10^{2.5(\Omega+0.5)} & for\ -1.3 \le \Omega \le -0.5 \\ 1 & for\ -0.5 \le \Omega \le 0.5 \\ 10^{-1.0(\Omega-0.5)} & for\ 0.5 \le \Omega \le 2.5 \\ 0 & for\ \Omega \ge 2.5 \end{cases}$$

The frequency boundaries of each of the critical bands of the filters are warped using Schroeder's Bark (z) to Frequency (f) equation:

$$f = 600\sinh\left(\frac{z}{6}\right)$$

This is equivalent to warping the power spectrum P(w) along its frequency axis into Bark frequencies, and then applying the auditory filters directly, rather than inverse warping them.

Next, Hermansky approximates the non equal sensitivity of human hearing at different frequencies using the following equation:

$$E(\omega) = [(\omega^2 + 56.8*10^6)\omega^4]/[(\omega^2 + 6.3*10^6)^2(\omega^2 + 0.38*10^9)(\omega^6 + 9.58*10^{26})]$$

In human hearing, the stimulus at one frequency will be perceived as louder than another stimulus of the same intensity that is at a different frequency. The resulting auditory filter bank is then convolved with the power spectrum P().

$$\Xi(\Omega(\omega)) = E(\omega)\Theta(\Omega(\omega))$$

An approximation to the power law of hearing is next applied, which is computed by:

$$\Phi(\Omega) = \Xi(\Omega)^{0.33}$$

Finally, the pre-processed frequency spectrum () is transformed back to the time-domain, and linear prediction coefficients are extracted to yield an all-pole autoregressive model of the speech. This autoregressive property of the PLP features is similar to that of the LPC features.

Perceptual Linear Prediction coefficients are used quite often for automatic speech recognition. Speaker-dependent information is largely stripped away, leaving the linguistic content, given that the autoregressive model is of a low order. Hermansky has reported that PLP coefficients of order 5 yields these characteristics best. Higher orders contain speaker-specific as well as lexical content[13].

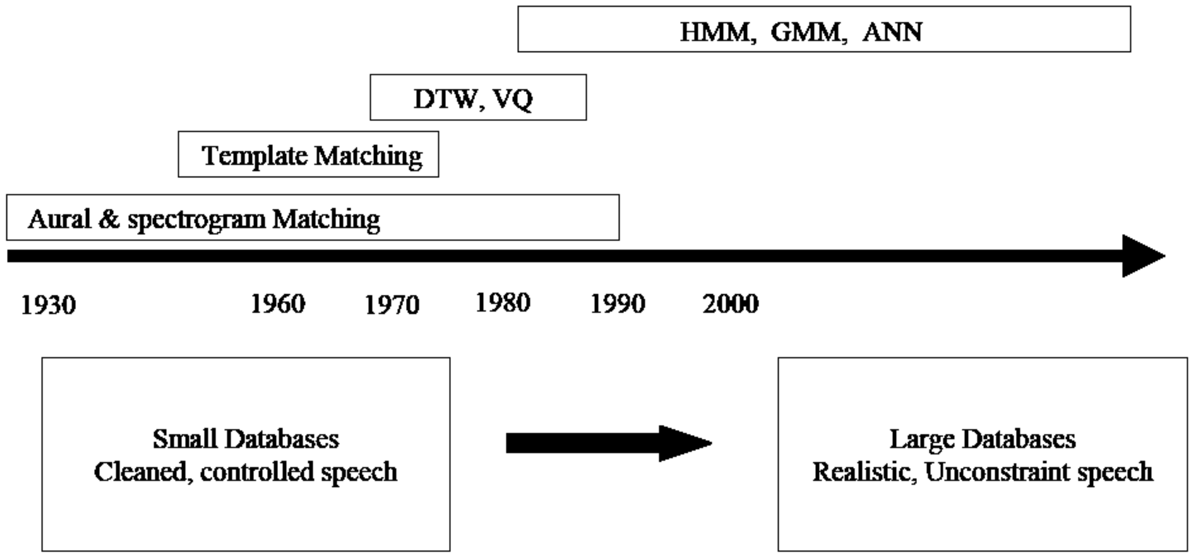
CLASSIFIERS

In the figure below, a time line is shown that depicts the shifting trends in the choice of classifiers for speaker recognition. As shown, there has been a shift from template classifiers to statistical classifiers in response to a shift from smaller databases recorded under more pristine laboratory conditions to larger databases recorded under more realistic, spontaneous, and noisy conditions.

This general shift from template to statistical classifiers has resulted in marked improvements in speech and speaker recognition in recent years. Not only are statistical models more robust under noise, but are more flexible in allowing for slight variations that were not seen in the training speech data. As already discussed, for speech recognition applications especially, there are usually too many variations to be able to adequately capture within the training set. It would take enormous amounts of input data in order to do a half-decent job, and the computational expense is unrealistic.

Statistical Classifiers

Statistical classifiers are better-suited to realistic speaking environments and have gained much popularity in the last couple of decades. It has been shown [6,11] that Gaussian Mixture Models (GMMs)[14] are one of the most flexible classifiers for text-independent speaker recognition and therefore they are able to achieve higher recognition performance in a more realistic setting. This is because they are more able to model variations in the speaker's voice samples than other classifiers, and are also more able to take into account noisy environments. Other statistical classifi-



ers, such as pdf-based methods, only make use of a single pdf component, whereas the GMM makes used of several, thereby allowing it to more tightly model a broader range of voice characteristics. Hidden Markov models (HMM) and HMM hybrids model temporal information, and are therefore better suited to text-dependent speaker recognition, in which pattern-matching the lexical content of an utterance is more important. For text-independent tasks, GMM's are a better choice than HMM's[6,11,14].

GAUSSIAN MIXTURE MODEL (GMM)

GMM classifiers use probabilistic measurements to determine class membership, or how "likely" it was that the class generated the new observation. The Gaussian probability distribution (pdf) is a common statistical classifier used for speech and speaker recognition. Given a set of data, it is possible to estimate the parameters for a Gaussian pdf that best fits the data.

A single Gaussian pdf describing a k-dimensional random variable X has the form:

$$g(x) = \frac{1}{(2\pi)^{\frac{k}{2}}|C|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu)^T C^{-1}(x-\mu)\right)$$

where C is the covariance matrix, |C| is its determinant, and is the mean vector.

An more sophisticated extension of the Gaussian pdf is the Gauss mixture pdf. A Gauss mixture pdf is a weighted sum of a collection of distinct Gaussian pdf's.

An n-variate Gaussian density is defined as:

$$b_i(x) = \frac{1}{(2\pi)^{\frac{k}{2}}|C_i|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x-\mu_i)^T C_i^{-1}(x-\mu_i)\right)$$

$$p(x|\lambda) = \sum_{i=1}^{M} p_i b_i(x),$$

constrained by

$$\sum_{i=1}^{M} p_i = 1$$

and where $b_i(x)$, i=1,...,M are the component densities and p_i is the weight of the i^{th} mixture component.

When used as a classifier, this is referred to as a Gaussian Mixture Model (GMM), where each training observation is interpreted as having been generated by one of the Gaussians. The Gaussian mixtures are built around the input training data in a best-fit manner. This is accomplished using the Expectation-Maximization (EM) algorithm [10] in an iterative fashion to estimate the parameters of each Gaussian.

GMM's are used in both speech and speaker recognition. In speaker recognition, they are commonly used in text-independent applications because of their flexible nature and their ability to take into account slight variations in the data, as well as noise. In speech recognition they are commonly used in conjunction with HMM's.

A CONFERENCING SYSTEM USING SPEAKER ID

Experiments we have performed have shown us two things. First we need to see if the PLP-LSP features outperform the traditional PLP features in a text-independent speaker recognition task. Second it would be good to know how the various methods compare against each other. Specifically, since PLP features outperform LPC features, do PLP-LSP features also outperform LSP features?

The table below shows the percent correct identifications for 44 people from the YOHO[1] database using 20th-order LPC, LSP, PLP, and PLP-LSP features.

LPC	88.7	LSP	98.2
PLP	94.0	PLP-LSP	97.7

A 98% success rate sounds pretty good overall, but the utility of the LSP scheme in this application will depend on the circumstances under which it fails and the nature of the failure. If the recognizer fails for short (under one second) periods at random, it is better than if it fails to determine that the speaker has changed. Also, recording a failure to recognize is better than recording that the wrong person is speaking.

In the conferencing system design a main principle is that the state (I.E. who is speaking) does not change unless a positive recognition occurs for a significant time, perhaps a few seconds. The speaker shift can be recorded in a log file at the time it actually occurred rather than when it was accepted. This resolves most potential problem situations.

For example, let's imagine that the system activates a camera pointing at the person speaking at the moment. If two people are speaking at the same time, the system will keep the camera active for the first person to speak. If the speaker stops, or takes a drink, or if the recognizer fails on an utterance, the speak does not change. Silence and unrecognized sounds do not change the speaker. Only when an utterance is recognized as having been made by one of the speakers present do a state change occur.

How quickly can the system assign an utterance to a speaker? Unfortunately it depends to some degree on the utterence. However, the recognition will take a few seconds, meaning that a camera shift that depends on speaker identity would be a few seconds behind the speech. For transcription, the system would do a backward search from the time of speaker change detection to the moment when the speaker actually changed and annotate from that point.

Entrainment is a small issue. The speaker recognition system must learn how to identify the speakers. In ongoing situations this is not really a problem, because training can occur off-line or during a previous conferencing session and then saved. We begin a new session by telling the system who is at the meeting, and the training data can be easily recalled. For the basic system in ambient circumstances it took at least 10 input utterances to do a decent job training on 3 or 4 real voices. It took only 10 to do recognition on the TIMIT[1] database, which was set in more pristine conditions. This would be a minute or so of speech, and at some point the system would have to be given names to associate with the speech samples that it had encountered. An operator could be brought in for the opening minutes of a meeting if there were people attending who had not been previously sampled.

CONCLUSIONS

A 98% success rate at recognition means a basic 59 minutes per hour of correct state. In fact it is better than that, because more than half of the failures will be dealt with properly by simply ignoring them. This means that we expect one minute every two hours of an incorrect ID.

ACKNOWLEDGMENTS

Thanks to Westgrid Visualization for help and funding.

REFERENCES

1. Campbell, J. & Reynolds, D. (1999, May 15-19). Corpora for the Evaluation of Speaker Recognition Systems. Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP-99), Phoenix, Arizona, 2247-2250.
2. Campbell, J. P. Speaker Recognition: A Tutorial. Proceedings of the IEEE, 85(9) Sept. 1997.
3. Campbell, J. P.& Tremain, T. E. (1986). Voiced/Unvoiced Classification of Speech with Applications to the U.S. Government LPC-10E Algorithm. Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 473-476.
4. Hermansky, H. (1990, April). Perceptual Linear Predictive (PLP) Analysis of Speech. Journal of the Acoustical Society of America, 87(4), 1738-1752.
5. Itakura, F. (1975, April). Line Spectrum Representation of Linear Predictive Coefficients of Speech Signals. Journal Acoustical Society of America, 57, p. S37.
6. Kasuriya, S., Wutiwiwatchai, C., Achariyakulporn, V., & Tanprasert, C. (2001). Comparative Study of Continuous Hidden Markov Models (CHMM) and Artificial Neural Network (ANN) on Speaker Identification System. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 9(6), 673-683.
7. Lu, N. A., Morrell, D. R. (1991, April 14-17). VQ Codebook Design using Improved Simulated Annealing Algorithms. International Conference on Acoustics, Speech, and Signal Processing (ICASSP-91), 1, Arizona State University, Tempe, Arizona, 673-676.
8. Makhoul, J. (1975, April). Linear Prediction: A Tutorial Review. Proceedings of the IEEE, 63(4), 561-580.
9. Makhoul, J. (1975, June). Spectral Linear Prediction: Properties and Applications. IEEE Trans. Acoust, Speech, Signal Processing, ASSP-32(3), 283-296.
10. Moon, T.K., The Expectation-Maximization algorithm, IEEE Signal Processing, 13(6) 1996. 4pp. 47-60.
11. Rabiner, L. R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE, 77(2), 257-286.
12. Reynolds, D. A., Quatieri, T. F. & Dunn, R. B. (2000). Speaker Verification Using Adapted Gaussian Mixture Models. Digital Signal Processing, 10(1-3), 19-41.
13. Ribeiro, C. M. & Trancoso, I. M. (1996, September). Application of Speaker Modification Techniques to Phonetic Vocoding. International Conference on Speech and Language Processing (ICSLP'96), 1, Philadelphia, 306-309.
14. RodrÌguez, E., RuÌz, B., GarcÌa-Crespo, ¡. & GarcÌa, F. (1997). Speech/Speaker Recognition Using a HMM/GMM Hybrid Model. AVBPA-1997, 227-234.
15. Soong, F. K. & Juang, B. H. (1984). Line Spectrum Pair (LSP) and Speech Data Compression. Proc ICASSP, 1.10.1-1.10.4. (Itakura, 1975)
16. Zheng, F., Song, Z., Li, L., Yu, W. & Wu, W. (1998). The distance measure for Line Spectrum Pairs applied to speech Recognition. Proceedings of the International Conference on Spoken Language Processing (ICSLP-98), 3, 1123-1126.

pcVOD: Internet Peer-to-Peer Video-On-Demand with Storage Caching on Peers

Lihang Ying and Anup Basu

Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada, T6G 2E8
Tel: (780) 492-3330, Fax: (780) 492-1071
{lihang, anup}@cs.ualberta.ca

ABSTRACT

Owing to its advantages of cheaply solving the server-side bottleneck, peer-to-peer (P2P) networking is gaining attention in recent years. At the same time, Video-On-Demand (VOD) is becoming much more popular on the Internet, such as online theatres and news websites. However, VOD consumes large bandwidth and traditional solutions can only serve very limited number of concurrent demands. P2P becomes useful in this situation. In this paper, we propose Peer Cache VOD (pcVOD), an Internet P2P VOD with caching on peers. pcVOD introduces the optimized management of resource sharing. As simulation results show, for demands on popular news video, with little cache space on each peer, pcVOD makes significant improvement compared to P2P VOD without the management of resource sharing.

1 Introduction

The soul of P2P is to share resources between peers and to utilize all the available resources on the Internet, where peers benefit from one other. P2P solves the bottleneck on the server under the centralized Client/Server architecture. Another advantage of P2P is low cost. P2P is an application-layer solution, which does not need upgrade to an existing network. It utilizes the resources of peers, which greatly reduces the requirements on the capability of a server, and in fact can make servers unnecessary. Owing to these advantages, P2P technologies are being adopted in both research and industry [1, 3, 4, 5, 12, 19].

With the wide adoption of high-speed Internet, multimedia streaming is becoming more and more popular. But one connection of multimedia streaming consumes much more bandwidth than traditional text based messaging. The server-side bottleneck constrains the scalability. In traditional Client/Server solutions, every demand is handled by a centralized server, requiring a powerful server and large bandwidth. Although server clusters can improve the processing ability, there is a limit on the bandwidth of a server. Because VOD streaming requires significant bandwidth, Client/Server solutions cannot support large number of demands over the Internet. For example, with 100Mb bandwidth and a 200Kbps video, the limit is only 500 concurrent demands, which is far from supporting VOD of hot events, such as the Olympic Game or popular news video. One solution is to setup proxy servers to cache video [2, 7], which is an expensive option and still not well scalable. P2P architecture mitigates the problems in this situation. In recent years, substantial research on application-layer multicasting and streaming with P2P has been conducted [6, 8, 10, 11, 15].

Multimedia streaming can be categorized into live streaming and on-demand streaming. For live streaming, all the clients are synchronous. For on-demand streaming, mostly clients demand different video or different parts of the same video. For popular video, which many clients demand at the same time, some methods, such as batching or patching [18], are proposed. These approaches work only if one video is demanded simultaneously. Less popular videos still consume limited server bandwidth. Another problem is that there is only single path from server to client. If the connection is slow or unstable, service quality will be bad. Although proxy-caching solution [2, 7] can deal with this situation to some extent, it is expensive and not very scalable. P2P VOD [13, 14, 16, 17, 20-27] was conceived to solve this problem. In this paper, we propose pcVOD, an Internet P2P VOD with storage caching on peers. First, instead of depending on user to manually decide how much cache space is shared and which video to be kept to share or not, our system required peers provide storage cache space and lets our cache replacement algorithm to manage cache. The cache replacement strategy is based on the ratio of the number of peers demanding to the number of peers with cache, which is unique in a P2P VOD environment. Second, instead of assigning task to peers by equal-size block, we assign as much as possible to one peer at a time while guaranteeing the data for current rendering is ready by bandwidth monitoring of connection to each peer.

This remainder of this paper is organized as follows: Section 2 describes the architecture and model. System details are given in Section 3. In Section 4, we perform simulations to verify the proposed approach. Real world implementation is described in Section 5. Concluding remarks are given in Section 6.

2 Architecture and Model

2.1 Architecture

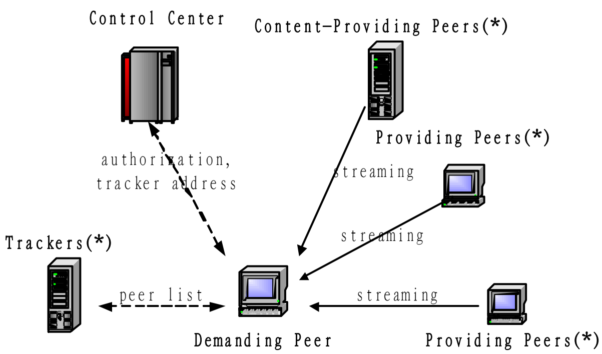


Figure 1: Architecture

Figure 1 shows the architecture of pcVOD. The motivation of this paper is to improve the BitTorrent-like P2P file sharing systems [3] to support legal commercial on-demand streaming. In order to charge from users for commercial applications, user authentication is critical. pcVOD has one centralized and reliable Control Center for user and node authentication and content-directory service. Although totally decentralized P2P systems [1] are fashion technology currently, they are complex and inefficient for networking, and more suitable for the situations without copyright and user authentication consideration. And also totally decentralized P2P systems have the risk that service is not available in case all peers with the resource are not online.

In pcVOD, there are content-providing peers and demanding peers. Always online content-providing peers are introduced to avoid the "no-seeds" problem in the P2P systems where unreliable normal peers provide contents, such as BitTorrent. And only authorized content-providing peers are can release new videos. Thus, the pirate problem of copyrighted video can be better controlled. Content-providing peers with large bandwidth garuantee the streaming bitrate of new released video and lack-of-cache video. Each demanding peer has storage space to cache videos. Cached video on normal peers is encrypted to prevent illegal spread. For implementation simplicity and networking efficiency, like BitTorrent[3], we use reliable Trackers to manage the connections of peers and caching. There are multiple Trackers to manage the connections and cache of different video. When a peer demands a video from the video list at the Control Center, the address of Tracker with the demanded video will be obtained from the URL of the video list. Then the peer sends a request to the tracker, which returns a list of candidate peers with the video cached. Then, video data is retrieved simultaneously from the candidate peers.

In our architecture, for considerations of commercial using, simplification, networking efficiency and reliability, always-online reliable Control Center and Trackers handle the processing requiring light burden and small bandwidth. P2P is only introduced in the most needed part, video data transmission, which has heavy burden and is bandwidth-consuming.

2.2 Storage Cache and Replacement Strategy

In BitTorrent-like P2P systems, users decide which files are going to be shared, regardless of whether the files are really needed by other peers. Here, we introduce storage management to improve the efficiency of storage space in P2P systems. Each peer is required to share certain size of hard-drive storage cache. (This assumption is reasonable if the required cache space is small or in applications like TV Set-top Box, in which large hard-drive storage cache can be integrated.) Also, our algorithm manages how to use space. When the shared space is full, the replacement algorithm is used to determine which stored video should be replaced. The replacement algorithm is described as follows.

In a P2P VOD environment, the greater the number of available peers with video cache, the higher the probability that the video is available, and the higher the effective bandwidth. We want a video with high demanding ratio to have more cache available. Thus, we design a rank of the cache requirement, *cache_needed_rank,* which is the ratio of the number of peers demanding to the number of online peers with cache.

$$cache\_needed\_rank = \frac{the\ number\ of\ \ peers\ demanding}{the\ number\ of\ online\ peers\ with\ cache}$$

The higher the *cache_needed_rank*, the higher is the priority of a stream being cached. The cache replacement policy is based on *cache_needed_rank*. The cached stream with a lower *cache_needed_rank* is replaced by one with a higher *cache_needed_rank*. The cache strategy takes into account both the video popularity and the number of peers that currently are caching that video. It is unique in the P2P environment, and is different from traditional caching schemes, such as Least-Recently-Used, aging, and popularity-based.

2.3 Pre-fetching Retrieval Strategy and Model

In pcVOD architecture, there is a hard-drive storage for cache. Thus, differing from usual streaming, it can fetch data in advance (pre -fetching) and store in storage. Also differing from traditional streaming from one server, it retrieves from many peers. Instead of fetching bits in order as in traditional streaming, pcVOD does not retrieve in exact bit-order. The retrieval procedure is divided into many pre-fetching periods. In each pre-fetching period, data is not fetched in order, but it is assigned to the connections of candidate peers block by block. The retrieval procedure pre-fetches as much data as possible while guaranteeing the data for current rendering is ready. With this strategy, we assign task to peers in larger blocks and thus reduce the protocol load. The detailed retrieval strategy is as follows:

1) At the beginning of streaming, set an initial pre-fetching size, which is determined by initial buffering time.
2) Assign one retrieving part to each connection of candidate peer with equal size.
3) When a connection to one peer finishes transmitting its assigned part, a new part from the unfinished current assignments of other connections is re-allocated, whose size is determined according to the statistically estimated bandwidth of this connection.
4) If the video is not all fetched, according to the sum of the statistically estimated bandwidth of each connection, a new pre-fetching size is computed and a new pre-fetching period starts.
5) Assign new pre-fetching task to each connection according to its current bandwidth and go to Step 3.

During retrieval, the new joining peers with the video cached and new peers finishing caching the whole video, will be added into candidate peers list. Also, the departing peers and slow peers will be deleted. Our retrieval method deals well with network bandwidth fluctuations, since the retrieval procedure monitors each connection's bandwidth, assigns tasks dynamically, and have relatively large hard-drive storage as buffer.

Every time a new retrieving task to a candidate peer is assigned, there is a control message sent. In order to reduce the protocol load, a lower number of task assignments is preferred. Obviously, the larger the size of each retrieval task is, the less the number of task assignments is. However, if the size of pre-fetching task is too large, it is possible that later data is being pre-fetched while the earlier data for current rendering is not ready. Pre-fetching size should be as large as possible while guaranteeing the data for current rendering is ready. The safe maximum pre-fetching size is equal to the estimated bandwidth times the safe pre-fetching time.

Figure 2 shows pre-fetching of a video file. The x-axis is the rendering time of the video. The gray part is the fetched data and the white part is the data to be fetched. T_r is the time point of current rendering and T_f is the time of the end of fetched data. Before rendering time reaches the end of fetched data, there is a time period of $(T_f - T_r)$. In order to guarantee the data for rendering at time T_f is ready, the pre-fetching must finish in the time period of $(T_f - T_r)$, which is the safe pre-fetching time.

Bandwidth is estimated by interval estimate using student's t-distribution [9]. After the bandwidth μ_{est} is estimated, the safe maximal pre-fetching size is equal to the estimated bandwidth μ_{est}, times the safe pre-fetching time, $(T_f - T_r)$.

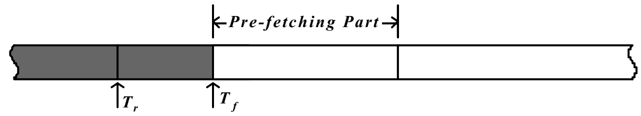


Figure 2: Diagram of Pre-fetching

Thus, pre-fetching size V is: $V = (T_f - T_r)\mu_{est}$

3 System Details

3.1 Data Structure

To manage the connections of peers and caching, there is a specified Tracker. Tracker maintains a table of Video Cache Information (*VCI*) of each video (Data Structure 1).

VCI includes video's id, *video_id,* its bitrate, *bitrate*, the number of peers demanding the video, *demanding_number*, the number of online peers with the video cached, *online_cache_number*, its rank of the cache requirement, *cache_needed_rank*, and a table of current caching information, *cache_info*. *cache_needed_rank* is the ratio of the number of peers demanding to the number of online peers with the video cached. *cache_info* includes caching peer's id, *peer_id*, and the online status, *is_online*.

Data Stucture 1:

```
stucture {
    int video_id;
    float bitrate;
    int demanding_number;
    int online_cache_number;
    float cached_needed_rank;
    structure {
        int peer_id;
        bool is_online;
    } cache_info[];
} VCI;
```

3.2 Main Procedure

The demanding procedure is as follows (Algorithm 1). In order to reduce the burden on Tracker, *cached_needed_rank* for the decision of cache replacement is requested only at the beginning.

Algorithm 1:

P gets Tracker address from video URL at Control Center;
P sends required video ID to Tracker and Tracker updates demanding_number of VCI and returns VCI to P;
P connects to candidate peers given in returned VCI

```
to retrieve the video;

caching_video = true;
while (retrieval is not finished && demanding is not
cancelled) {
      if (the cache space is full && caching_video
== true) {
            pick the video with the lowest
            cache_needed_rank in local cached video:
            local_cached_streams[l];

            if (video.needed_rank
                        <
            local_cached_streams[l].cache_needed_rank)
            {
                caching_video = false;
            }
            else {
                free the space of local_cached_streams[l];
                Tracker deletes local_cached_streams[l]
            in VCI;
            }
      };
      if (caching_video == true)
         cache the video;
};
if (caching_video == false) {
   free the space of partly cached video;
}
if (retrieval is finished and caching_video == true) {
   Tracker adds P into VCI;
}
```

4 Simulation Evaluation

4.1 Experimental Setup

We simulate a typical high-volume application, demanding short video of hot events in a news website. Since we focus on comparing different distribution and retrieval strategies, the simulation deals with the processing at the application level only and has not considered the low-level networking issues, such as topology. We implement the simulator ourselves using C++ language and can run on both Windows and Unix platform. Our simulations run on PC, with Pentium-4 2.4G CPU, 1G Memory, and Linux Operating System.

The parameters of the simulation are designed considering the practical situations and the balance between making the simulation possible and avoiding loss of generality. As shown in Table 1, the simulation has 10,000 peers and 206 videos. The samples are from all available news videos on May 23, 2004 at CNN website (http://edition.cnn.com/video/). Figure 3 shows snapshots of some of the videos used. The bitrate of the videos is 37.6KBps (301Kbps) and the duration ranges from 37 to 384 seconds with a mean of 140.0 and standard deviation of 65.7.

Figure 3: Snapshots of sample videos used.

Table 1: Simulation Setup

Number of Peers	10,000
Online Peers	2,000
Number of Videos	206
Bitrate of Videos	37.6KBps
Mean of Duration of Videos	140.0 seconds
Std. Dev. of Duration of Videos	65.7
Up-Link Bandwidth of Content-Provider Peer	2MBps
Up/Down-Link Bandwidth of Other Peers	50-200KBps
Rate of Peer Login	1 per second
Rate of Peer Logout	1 per second
Rate of Peer Demanding	10 per second

At the beginning, a tracker, one content-provider peer, and 2,000 other peers without any cache, are online. There is one peer logging out every second. The peers logging out are randomly selected from online peers. Also, there is one peer logging in every second. The peers logging in are randomly selected from offline peers. There are 10 peers demanding a video in one second. The demanding peers are randomly selected from online peers, which do not demand currently. Finally, the video demanded is randomly selected.

The up-link bandwidth of the content-provider peer is 2 MBps. The available bandwidth of each connection is chosen randomly in the range [50KBps, 200KBps]. The up-link and down-link bandwidth limitation of peers is chosen randomly in the range [50KBps, 200KBps]. When the upper-bound bandwidth on peers is reached, the bandwidth of each connection is assigned proportionally to the ideal bandwidth of the connection without competition from other connections.

4.2 Experimental Results

4.2.1 Comparison between pcVOD, P2P VOD without cache management, and Client/Server VOD

Owing to substantially different architectures and applications, we do not compare with Proxy Solutions. We compare pcVOD with P2P VOD without cache management, which has the same amount of total cache space in the system. Suppose pcVOD has a cache of C Mega-byte on each peer. In P2P VOD without cache management, 50% percent of peers have no cache, another 50% percent of peers have cache. The cache size is chosen randomly in the range [0, 4*C]. One of the cached videos is chosen randomly, to be replaced by a new video when the cache space is full. In the simulation, $C = 10$.

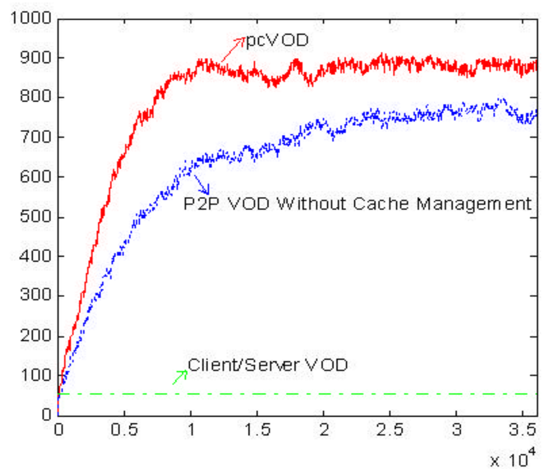


Figure 4: Comparison between pcVOD, P2P VOD without cache management, Client/Server VOD.

Since a P2P architecture mainly solves the problem of server bottleneck, we focus on the volume of demands served. In our simulation, a peer quits demanding if there is frequent rebuffering, which means pre-fetching is slower than playing. Thus, we use the number of demanding peers as evaluation criteria, which reflects the volume of demands served. Figure 4 shows the comparison between pcVOD, P2P VOD without cache management, and Client/Server VOD. The dash-dot line is the maximum number of demands served by a traditional Client/Server solution, which is equal to 53, since the up-link bandwidth of content-provider peer (2MBps) can only be shared by at most 53 clients when the bitrate of video is 37.6KBps. As Figure 4 shows, pcVOD and P2P VOD have much higher volume of served demands, which justify the benefits of P2P architectures. Also, pcVOD shows significant improvement compared to P2P VOD. The reasons are as follows: In P2P VOD, there are peers without cached video and peers with larger cache space and more video cached. As a result, while the up-link bandwidth of peers without cache is not utilized, more peers retrieve data from the peers with cache, which makes these peers reach the limitation of up-link bandwidth. In pcVOD, cached videos are distributed to more peers, better utilizing the bandwidth of more peers.

4.2.2 Analysis on pcVOD with different Cache Sizes

Figure 5 compares pcVOD with different Cache Sizes, 5M, 10M, 20M and 40M. It can be observed in Figure 5, that the larger Cache Size is, the more the volume of demands served is. However, the volumes for Cache Sizes of 20M and 40M, are almost the same. This means that after the Cache Size reaches a certain threshold, the volume of demands served stays almost the same. This happens because when the Cache Size reaches the threshold, there is enough number of caches of each video and pcVOD can serve every demand.

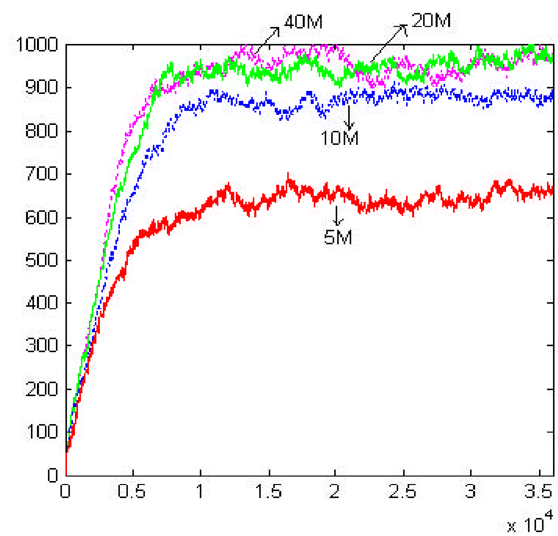


Figure 5: Analysis of pcVOD with different Cache Sizes.

5 Real World Implementation

Figure 6: Snapshot of peer in pcVOD demo system

We have implemented a prototype of pcVOD. A demo system is available at: http://pcvod.3322.org. In this demo system, the control center, one content-provider peer ('videodata.3322.org') and one normal peer ('218.5.76.98') with cached video locate in Xiamen, China, and one normal peer ('129.128.16.21') with cached video locates in Univ. of Alberta, Edmonton, Canada. Figure 6 shows a snapshot of an experiment with a demanding peer ('68.149.207.70') with cable high-speed access from a residence, at Edmonton, Canada.

The left is the main playbacking window of the demanding peer. The right-top is the window of the demanding peer tracking the candidate peers. To guarantee the sufficient streaming bitrate, multiple concurrent connections are setup to retrieve data from content-provider peer. In order to reduce the burden of a normal peer, only one connection is created to a normal peer. The right-bottom is the window of the peer '129.128.16.21', which tracks the uploading bitrate.

6 Conclusions and Future Works

In this paper, we proposed pcVOD, a P2P VOD with caching on peers. pcVOD introduces the management of sharing storage of peers in P2P VOD. Differing from common streaming, there is hard-drive storage at each peer for cache. The cache strategy is based on the ratio of the number of peers demanding to the number of online peers with cache. To reduce the protocol load, pre-fetching is proposed. The pre-fetching size is statistically calculated. As simulation results show, with little cache space on each peer, pcVOD shows great improvement compared to current P2P VOD. We also analyse the effect of the size of cache space on VOD serving capability of pcVOD. It is shown that optimum performance can be achieved for a reasonably small cache size.

In future work, we will conduct experiments on real-world systems to gather, compare and analyze the performance results. Through these real-world systems, we will track the behavior of peers in a typical Internet VOD application. For example, we will estimate the percentage of online peers during different time of a day; then, using this data, model and analyze of the behavior of peers and pcVOD.

References

[1] B.Y. Zhao *et al.*, Tapestry: A Resilient Global-scale Overlay for Service Deployment, IEEE Journal on SAC, Vol. 22, No. 1, Pgs. 41-53, 2004.

[2] B. Wang *et al.*, Optimal Proxy Cache Allocation for Efficient Streaming Media Distribution, IEEE Infocom 2002, New York, USA.

[3] B. Cohen, Incentives Build Robustness in BitTorrent, http://bitconjurer.org/BitTorrent/.

[4] D.S. Bernstein *et al.*, Adaptive Peer Selection, 2nd International Workshop on Peer-to-Peer Systems, Berkeley, CA, USA, 2003.

[5] D. Liben-Nowell *et al.*, Observations on the Dynamic Evolution of Peer-to-Peer Networks, Int. Workshop on P2P Systems, 2002.

[6] D. Xu *et al.*, On Peer-to-Peer Media Streaming, The 22nd International Conference on Distributed Computing Systems, Vienna, Austria, 2002.

[7] D.A. Tran, K.A. Hua, and S. Sheu, A New Caching Architecture for Efficient Video Services on the Internet, IEEE The 2003 International Symposium on Applications and the Internet, Orlando, Florida, 2003.

[8] H. Wang, G. Shen, S. Li, and Y. Zhong, Enhancing Multimedia Streaming Performance Through Peer-Paired Collaboration, IEEE International Conference on Image Processing, Rochester, NY, 2002.

[9] L. Ying, A. Basu, and S. Tripathi, Multi-Server Optimal Bandwidth Monitoring for Collaborative Distributed Retrieval, IEEE 2004 International Symposium on Circuits and Systems, Vancouver, 2004.

[10] M. Castro *et al.*, SplitStream: High-Bandwidth Content Distribution in Cooperative Environments, Int. Workshop on P2P Systems, 2003.

[11] M. Hefeeda *et al.*, PROMISE: Peer-to-Peer Media Streaming Using CollectCast, ACM Multimedia Berkeley, CA, 2003.

[12] T.S. Eugene Ng *et al.*, Measurement-Based Optimization Techniques for Bandwidth-Demanding P2P, IEEE Infocom, San Francisco, 2003.

[13] T. Nguyen and A. Zakhor, Distributed Video Streaming Over Internet, SPIE Multimedia Computing and Networking, San Jose, CA, 2002.

[14] X. Jiang *et al.*, Gnustream: A P2P Media Streaming System Prototype, IEEE ICME, Baltimore, Maryland, 2003.

[15] Y. Chu *et al.*, A Case for End System Multicast, IEEE Journal on SAC, Special Issue on Networking Support for Multicast, 2002.

[16] Y. Guo, K. Suh, J. Kurose, and D.Towsley, A Peer-to-Peer On-Demand Streaming Service and Its Performance Evaluation, IEEE ICME 2003.

[17] Y.Guo *et al.*, P2Cast:P2P Patching Scheme for VoD Service, the 12th International World Wide Web Conference, Budapest, Hungary, 2003.

[18] Z.Xiang *et al.*, PeriodPatch: An Efficient Stream Schedule for VoD, SPIE Conf. on Internet Multimedia Mgmt. System, 2000.

[19] Sun Microsystems, Inc., JXTA Project, http://www.jxta.org/

[20] Yi-Cheng Tua *et al.*, Performance Analysis of a Hybrid Media Streaming System, SPIE Conf. on Multimedia Computing and Networking, Santa Clara, California, 1-22-2004.

[21] D. Qiu and R. Srikant, Modeling and Performance Analysis of BitTorrent-Like Peer-to-Peer Networks, SIGCOMM 2004.

[22] A.A. Hamra *et al.*, A Pull-Based Approach for a VoD Service in P2P Networks, 7th IEEE International Conference on High Speed Networks and Multimedia Communications, Toulouse, France, 2004.

[23] C. Loeser *et al.*, Distributed Video on Demand Services on Peer to Peer Basis, Int. Workshop on Real-Time LANs in The Internet Age, 2002.

[24] W.J. Jeon *et al.*, Peer-to-Peer Multimedia Streaming and Caching Service, ICME2002.

[25] C. Chan *et al.*, Peer-to-Peer Video Delivery Scheme for Large Scale Video-On-Demand Applications, ICME 2004.

[26] T. Do *et al.*, P2VoD: Providing Fault Tolerant Video-on-Demand Streaming in Peer-to-Peer Environment, International Conference on Communications 2004.

[27] S. Jin *et al.*, Cache-and-Relay Streaming Media Delivery for Asynchronous Clients, International Workshop on Networked Group Communication (NGC) 2002.

Disk Performance and VBR Admission Control for Media Servers

Dwight Makaroff Jason Coutu Fujian Liu

Department of Computer Science
University of Saskatchewan
Saskatoon, SK, S7N 5C9, Canada
{makaroff, jdc135, ful113}@cs.usask.ca

Abstract

Streaming applications have used many different approaches to provide delay-sensitive data to clients across networks. Previous work [4] showed that a detailed deterministic-guarantee admission algorithm can achieve close to optimal results with little overhead on commodity hardware. Changes in hardware technology may influence the nature of admission control results.

We evaluate admission performance based on bandwidth and buffer space limitations on multiple deployments of the server. The admission control algorithm used in this work can accept scenarios that utilize a high percentage of the available disk bandwidth when the average disk performance is close to the worst case performance. Increasing variability in disk bandwidth makes admission control decisions more conservative, but modifying a parameter of the algorithm helps achieve high acceptance rates.

Keywords*: admission control, variable bit-rate video, disk performance, system support*

1. Introduction

The delivery of stored video content has received much attention in the research and commercial communities. Client concerns have been CPU resources for decoding, as well as download bandwidth. Early systems supported a fixed number of constant bit-rate (CBR) streams [9]. RAID disk subsystems were necessary to get sufficient bandwidth for video. Over-provisioning both the disk and network to support variable bit-rate (VBR) was also one early technique.

There have also been research projects concerning admission control, ranging from statistical admission control [10, 11] to more advanced deterministic guarantees [2, 5]. None of these have been widely deployed, due to the scope/nature of the projects and the risk in having complicated mechanisms that may not be economically beneficial.

In this work, we consider high-quality, variable-bit-rate video streams that are of short playback duration, such as for news and/or music-video-on-demand environments that can be supported by a system composed of commodity personal computers. Substantial changes have taken place in the resource capabilities of this class of computers since the first generation of such systems. For a given cost, disk capacity, RAM size and CPU speeds have increased by an order of magnitude or more.

The increased variability of the disk subsystem has substantial influence on the total bandwidth that can be guaranteed from the server. If the average performance of the disk is less than 50% better than the worst case, our admission control has good results. With highly variable disk performance and substantial sequential reading, the performance guarantee is reduced. When the *minimum average past performance* is used as the worst case bandwidth estimation, however, the system achieves high utilization of the disk.

The remainder of this paper is organized as follows. Section 2 describes the system model. In Section 3, we provide the context of the experiments measurements. The results obtained for MPEG-2 video data clips are given in in Section 4. Section 5 describes the related work, while we conclude in Section 6 with indications for future work.

2. System Model

The Continuous Media File Server (CMFS) [5] is organized as a distributed system, with an administrator node and several server nodes. Server nodes can be added until the bandwidth of a particular network interface is saturated. Our admission control algorithm is called *vbrSim* and has been described and evaluated in other previous work [4, 6]. The main parameter to *vbrSim* is an empirically-derived measure [5] of the guaranteed disk bandwidth, hereafter called *minRead*. Data is read in a periodic fashion. During each time period (called a *slot*[1]), the disk is guaranteed to read at least *minRead* blocks in an earliest-deadline-first manner.

This assumption has the effect of reducing requirements in future slots, should a present slot's requirements be less than *minRead*. The CMFS will read faster than required, and store the data in server buffers. This uses slack bandwidth to smooth even more peaks in the future requirements. In the CMFS, the user can choose starting points (*start*/*stop* parameters) and fast motion or slow motion (*speed* parameter). High-speed scan delivers a subset of the frames (*skip* parameter). Each parameter can be independently modified by the user. The interaction of these parameters changes the VBR profile of the stream, and thus, the disk requirements, so that delivery schedules cannot be precomputed.

Buffers hold blocks that are read in advance of their deadline. A modest amount of buffer space can be allocated that corresponds to non-trivial playback time per stream. For example a 500 MB buffer could hold 20 MB for each of 25 streams[2]. Any stream requests arriving during steady state would read contiguously to catch up. This sequential reading provides great bandwidth improvement.

3. Experimental Environment and Design

The server nodes have AMD-Athlon processors (dual-booting FreeBSD 1.6 and Intel Solaris 2.8), with SCSI disks attached via SCSI-Ultra-160 buses (see Table 1). There are 4 basic hardware configurations for the experiments: Solaris-small disk, Solaris-big disk, FreeBSD-small disk and FreeBSD-big disk. The experiments were performed once per configuration. A small number of scenarios were repeated multiple times to confirm the stability of the achieved bandwidth.

Characteristic	Big Disk	Small Disk
Manufacturer	Seagate	Quantum
Brand	ST318406LW	Atlas IV
Capacity (GBytes)	18	9
RPM	10000	5400

Table 1: Disk Characteristics

Each disk was loaded with medium-length video clips (2 to 12 minutes), typical of news-on-demand or music-on-demand scenarios. The average bit rate varied from 1.46 Mbps to 8.55 Mbps, with an average of 5.55 Mbps. All streams were encoded using a software MPEG-2 encoder, with VBR encoding at 30 fps at a resolution of 640×480.

Request scenarios are randomly generated. First, the file to be requested is selected. Then, the portion of the file, the direction of desired playback, *speed* and *skip* are selected modeling varying modes of user requests.

Four arrival patterns are examined for the small disk: unique requests with mean inter-arrival time of 500 ms and 1000 ms, and repeated requests at 500 ms arrivals and 1000 ms. The arrival pattern of the requests is Poisson, with the inter-arrival time set to capture different traffic intensities. With this traffic intensity range, all requests arrive within the first minute and no more can be accepted until at least one stream has completed service.

The experiments use a server with unlimited buffer space in order to isolate the bandwidth effects from the limitations imposed by buffer space. Thus, longer delays between arrivals (i.e. lower traffic intensity) will result in the server buffering large amounts of each stream before the next request arrives, leading to decreased parallel reading. In the extreme, this becomes downloading. If enough *buffer space* is provided to enable acceptance of rapidly occurring requests, then scenarios in which requests arrive more slowly will have the *bandwidth* to support the request.

We consider the Cumulative Average Bandwidth Requested (CABR) as a percentage of the bandwidth achieved for that particular scenario (BA) The number of streams is a meaningless quantity; different streams have different total/average bandwidth requirements. Total bandwidth requested is also unsatisfactory, because a request for B Mbps may or may not be supported by the disk system, depending on the number of seeks and the data location. Scenarios with the same percentage of disk requests (CABR/BA) are hereafter referred to as a *request band*.

Bandwidth measures are estimates. *Request* bandwidth is the sum of the average bandwidth of each stream. This is an *overestimate* of the actual bandwidth, due to the varying stream playback length, but is accurate when all streams are actively using the disk. *Achieved* bandwidth is also an estimate. The minimum stable value of the average bandwidth is used as the measure of achieved disk performance.

High accept rates for scenarios in request bands near 100% of disk bandwidth capacity is a desirable outcome. Acceptances above 100% are possible, due to the high bandwidth available at the beginning of the scenario.

4. Results

4.1. Disk Bandwidth

The results for minimum bandwidth calculations and minimum observed behaviour is shown in Table 2. A set of calibration programs were run on each configuration to determine *minRead*. These programs measured read times from pathological disk block locations. FreeBSD provides a guaranteed minimum bandwidth slightly less than 10% better than Solaris. The situation is reversed for the large disk,

1 In the experiments, the slot time is 500 ms
2 25 seconds of 6 Mbps MPEG-2 video

where Solaris exceeds FreeBSD by 16%. Since the exact same disks were used, this difference is perplexing.

For the small disk on Solaris, the bandwidth achieved was on average 142% of *minRead* (64 blocks per slot), but the range varied due to block location and amount of sequential reading. The average performance on FreeBSD is similar. The difference between observed minimum and guaranteed minimum is much greater on the large disk. The lowest observed value on FreeBSD is 2.2 times *minRead*. The difference in performance between operating systems is part of future work. This will determine if there are OS-level optimizations that can significantly improve performance.

Disk	**FreeBSD**		**Solaris**	
	min	**Observed**	**min**	**Observed**
9 GB Quantum	50	60 to 82	46	54 to 73
18 GB Seagate	78	172 to 205	91	161 to 206

Table 2: Average Disk Performance

4.2. Small Disk

Figure 1 shows the admission results for Solaris and FreeBSD with unique stream requests. The results for the scenarios with repeated requests are similar. We can see that the percentage of the scenarios accepted decreases as percentage of bandwidth requested approaches 100%. Since relatively few scenarios could be supported by the disk at a rate greater than 100%, the results for the high request bands are not as reliable. Neither FreeBSD nor Solaris could accept many scenarios which requested over 85% of the disk bandwidth capacity at either arrival rate.

When an additional stream is requested, it is most often the case that the disk system cannot support the request. A new request increases the request percentage nearly 10%. Moving from 80% to 90% shows a decrease in acceptance rates, but moving from 90% to 100% shows a increase in the failure of the disk system to deliver the data.

Recall that the minimum of observed performance for these scenarios was 54 blocks/slot on Solaris and 60 on FreeBSD. If we set *minRead* to the minimum observed value of disk performance in the recent past, there is an increase in acceptance rates. These configurations accept nearly all scenarios requesting below 92% of disk bandwidth capacity.

Buffer space usage results are fairly consistent with the previous work. All the accepted scenarios require fewer than 10000 buffers (or 640 MBytes). This amount of memory to dedicate for buffers is available on desktop machines, but is rather large. Over 99% of the scenarios can be accepted with 6000 or fewer buffers (less than 400 MBytes), a reduction of 40% from the worst case amount of buffer space, with a small decrease in acceptance rates.

4.3. Large Disk

On the larger disk, the results are not as encouraging. The same shape of acceptance rates is observed. Most streams below 50% of disk bandwidth capacity are accepted, with a steady decrease between 50% and 60%, with only a few acceptances beyond 60%, even with extended inter-arrival times of 1500 ms and 2000 ms. This greatly under-utilizes the disk bandwidth and is not shown.

When *minRead* is set to the minimum *observed* bandwidth, the admission control results improve dramatically. On FreeBSD, the value used was 149, based on the lowest observed bandwidth in any *individual* slot in any of the scenarios, and therefore lower than any of the minimum cumulative averages. This is also almost twice *minRead*. The results are shown in Figure 2. Note that this is very similar to the small disk results with *minRead* equal to the observed minimum, in that almost all scenarios requesting less than 92% of disk capacity are accepted. Most scenarios with more than 100% of disk capacity are rejected as they cannot be supported.

The use of buffer space on the large disk is similar to that for the small disk. It is not shown, due to space constraints. The overall shape is the same, but the range is larger. Most scenarios between 5000 and 20000 buffers (320 MBytes to 1.3 GByte) of buffer space for a single disk. This is large, but not excessive for a single disk video server.

5. Related Work

One method of providing increased robustness in the bandwidth and increased flexibility in the scalability of the server has used data layout techniques, such as randomized placement on heterogeneous disks [7]. The Yima Continuous Media Server [8] uses pseudo-random placement [3] and statistical admission control [11] to increase the flexibility for variable delivery rates and interactivity. The admission control mechanism used by Zimmerman and Fu [11] explicitly considers writing and reading at the same time, but involves very complex mathematical formulations and complex disk modeling to get the most benefit out of the disk system. This disk modeling needs only be done once per disk, which is similar to our calibration method.

A similar system to the CMFS described in this paper is implemented and evaluated by Dimitrijevic *et al.* [2]. They have 3 components, an I/O Scheduler, a Request Scheduler, and an Admission Controller. The system uses conservative and aggressive versions of the admission control and achieves close to optimal results as well.

Server-based smoothing [1] shows an increase in the number of streams can be supported over non-smoothed streams, but such optimized algorithms require tens of seconds to smooth 30 minute clips. Our linear algorithm works on-line in less than a millisecond for 1 hour clips.

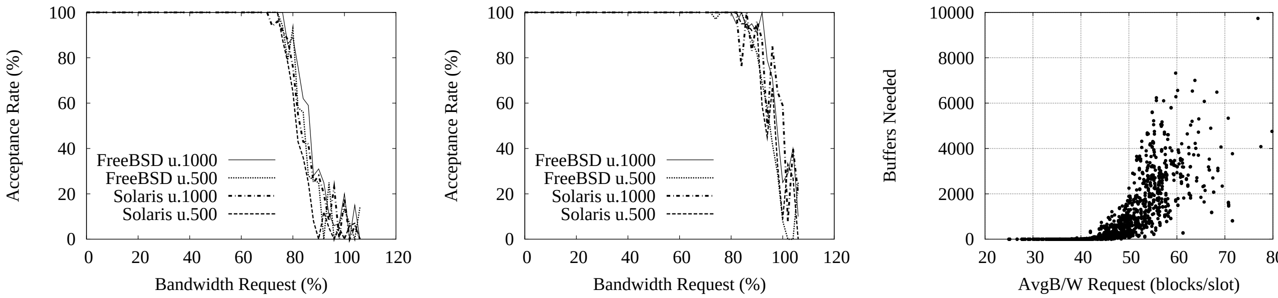


Figure 1: Small Disk Results

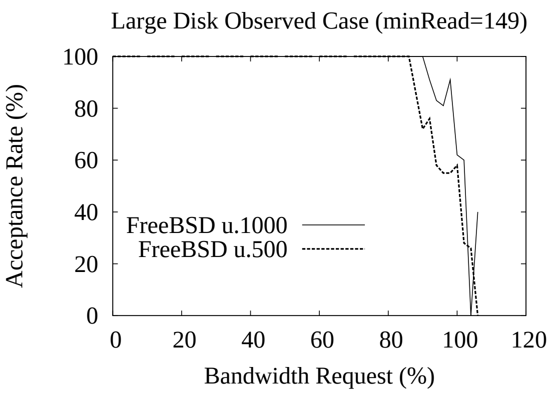


Figure 2: Admission for Large Disk

6. Conclusions and Future Work

The advances in performance of desktop disks and increased availability of memory for buffers have made it possible to provide video service with commodity components that can scale down to a modest single-disk video server.

The *vbrSim* admission control algorithm provided encouraging results several years ago. Experiments with more modern disks show that good admission results are still achieved when average and worst-case bandwidth guarantee are less than a factor of 1.5 apart. When the variability of the disk bandwidth is increased and the worst-case guarantee is used, the system greatly under-utilizes the available resources. Relaxing the value of *minRead* to the *observed* minimum provides good results in this case.

Future work will consider the differences in admission behaviour for feature-length streams and increased arrival rates. More detailed instrumentation of the server is necessary so that it can measure the resource usage of bandwidth and buffer space simultaneously.

Another issue to (re)examine is the data layout of streams on the disk. Our architecture does not make assumptions about the location on the disk, but contiguous allocation of media data for each object permits a great increase in bandwidth. Disk capacity is now another order of magnitude larger than the devices used in this study, with likely even more variability.

References

[1] S. V. Anastasiadis, K. C. Sevcik, and M. Stumm. Server-Based Smoothing of Variable Bit-Rate Streams. In *ACM Multimedia*, pages 147–158, Ottawa, ON, Canada, Sept. 2001.

[2] Z. Dimitrijevic, R. Rangaswami, and E. Chang. The xtream multimedia system. In *IEEE International Conference on Multimedia and Expo*, pages 545–548, Lausanne, Switzerland, Aug. 2002.

[3] A. Goel, C. Shahabi, S.-Y. Yao, and R. Zimmermann. SCADDAR: An Efficient Randomized Technique to Reorganize Continuous Media Blocks. In *Proceedings of the International Conference on Data Engineering*, pages 473–482, San Jose, CA, 2002.

[4] D. Makaroff, G. Neufeld, and N. Hutchinson. An Evaluation of VBR Disk Admission Algorithms for Continuous Media File Servers. In *ACM Multimedia*, pages 143–154, Seattle, WA, Nov. 1997.

[5] D. Makaroff, G. Neufeld, and N. Hutchinson. Design and Implementation of a VBR Continuous Media File Server. *IEEE Transactions on Software Engineering*, 27(1):13–28, Jan. 2001.

[6] D. Makaroff, G. Neufeld, and N. Hutchinson. Performance Evaluation for VBR Continuous Media File Server Admission Control. *Software Practise and Experience*, 34:1187–1210, July 2004.

[7] J. Santos, R. Muntz, and B. Ribeiro-Neto. Comparing Random Data Allocation and Data Striping in Multimedia Servers. In *ACM Sigmetrics*, pages 44–55, Santa Clara, CA, May 2000.

[8] C. Shahabi, R. Zimmermann, K. Fu, and S. Yao. Yima: A Second-Generation Continuous Media Server. *IEEE Computer*, 35(7):56–64, 2002.

[9] F. A. Tobagi, J. Pang, R. Baird, and M. Gang. Streaming RAID - A Disk Array Management System For Video Files. In *ACM Multimedia*, pages 393–400, Anaheim, CA, June 1993.

[10] H. M. Vin, P. Goyal, A. Goyal, and A. Goyal. A Statistical Admission Control Algorithm for Multimedia Servers. In *ACM Multimedia*, pages 33–40, San Francisco, CA, Oct. 1994.

[11] R. Zimmermann and K. Fu. Comprehensive Statistical Admission Control for Streaming Media Servers. In *ACM Multimedia*, pages 75–85, Berkeley, CA, 2003.

Assessment of Data Path Implementations for Download and Streaming

Pål Halvorsen[1,2], Tom Anders Dalseng[1], Carsten Griwodz[1,2]
[1]IFI, University of Oslo, Norway [2]Simula Research Laboratory, Norway
Email: {paalh, tdalseng, griff}@ifi.uio.no

Abstract

Distributed multimedia streaming systems are increasingly popular due to technological advances, and numerous streaming services are available today. On servers or proxy caches, there is a huge scaling challenge in supporting thousands of concurrent users that request delivery of high-rate, time-dependent data like audio and video, because this requires transfers of large amounts of data through several sub-systems within a streaming node. Since the speed increase for memory accesses does not follow suite with the CPU speed, copy operations can be a severe limiting factor on the streaming performance of off-the-shelf operating systems, which still have only limited support for data paths that have been optimized for streaming despite previous research proposals. We observe furthermore that while CPU speed continues to increase, system call overhead has grown as well, adding to the cost of data movement. In this paper, we therefore revisit the data movement problem and provide a comprehensive evaluation of possible streaming data I/O paths in Linux 2.6 kernel. We have implemented and evaluated enhanced mechanisms and show how to provide support for more efficient memory usage and reduction of user/kernel space switches for streaming applications.

1 Introduction

Improvements in access network connectivity with flatrate Internet connections, such as DSL, cable modems and recently E-PON, and large improvements in machine hardware make distributed multimedia streaming applications increasingly popular and numerous streaming services are available today, e.g., movie-on-demand (Broadpark), news-on-demand (CNN), media content download (iTunes), online radio (BBC), Internet telephony (Skype), etc. At the client side, there is usually no problem presenting the streamed content to the user. On the server side or on intermediate nodes like proxy caches, however, the increased popularity and the increasing data rates of access networks make the scaling challenge even worse when thousands of concurrent users request delivery of high-rate, time-dependent data like audio and video.

In such media streaming scenarios (and many others) that do not require data touching operations, the most expensive server-side operation is moving data from disk to network including encapsulating the data in application- and network packet headers. A proxy cache may additionally forward data from the origin server, make a cached copy of a data element, perform transcoding, etc. Thus, both servers and intermediate nodes that move large amounts of data through several sub-systems within the node may experience high loads as most of the performed operations are both resource and time consuming. Especially, memory copying and address space switches consume a lot of resources [1, 2], and since improvements in memory access speed do not keep up with the increase in CPU speed, these operations will be a severe limiting factor on streaming performance of off-the-shelf operating systems having only limited support for optimized data paths.

In the last 15 years, the area of data transfer overhead has been a major thread in operating system research. In this paper, we have made a Linux 2.6 case study to determine whether more recent hardware and commodity operating systems like Linux have been able to overcome the problems and how close to more optimized data paths the existing solutions are. The reason for this is that a lot of work has been performed in the area of reducing data movement overhead, and many mechanisms have been proposed using virtual memory remapping and shared memory as basic techniques. Off-the-shelf operating systems today frequently include data path optimizations for common applications, such as web server functions. They do not, however, add explicit support for streaming media, and consequently, a lot of streaming service providers make their own implementations. We investigate therefore to which extent the generic functions are sufficient and whether dedicated support for streaming applications can still considerably improve performance. We revisit this data movement problem and provide a comprehensive evaluation of different mechanisms using different data paths in the Linux 2.6 kernel. We have performed several experiments to see the real performance retrieving data from disk and sending data as RTP packets to remote clients. Additionally, we have also implemented and evaluated enhanced mechanisms and we can show that they still improve the performance of streaming operations by providing means for more efficient memory usage and reduction of user/kernel space switches. In particular, we are able to reduce the CPU usage by approximately 27% compared to best existing case removing copy operations and system calls for a given stream.

The rest of this paper is organized as follows: Section 2 gives a small overview of examples of existing mechanisms. In section 3, we present the evaluation of existing mechanisms in the Linux 2.6 kernel, and section 4 describes and evaluates some new enhanced system calls improving the disk-network data path for streaming applications. Section 5 gives a discussion, and finally, in section 6, we conclude the paper.

2 Related Work

The concept of using buffer management to reduce the overhead of cross-domain data transfers to improve I/O performance is rather old. It has been a major issue in operating systems research where variants of this work have been implemented in various operating systems mainly using virtual memory remapping and shared memory as basic techniques. Already in 1972, *Tenex* [3] used virtual copying, i.e., several pointers in virtual memory to one physical page. Later, several systems have been designed which use virtual memory remapping techniques to transfer data between protection domains without requiring several physical data copies. An interprocess data transfer occurs simply by changing the ownership of a memory region from one process to another. Several general purpose mechanisms supporting a zero-copy data path between disk and network adapter have been proposed, including Container Shipping [4], IO-Lite [1], and UVM virtual memory system [5] which use some kind of page remapping, data sharing, or a combination. In addition to mechanisms removing copy operations in all kinds of I/O, some mechanisms have been designed to create a fast in-kernel data path from one device to another, e.g., the disk-to-network data path. These mechanisms do not transfer data between user and kernel space, but keep the data within the kernel and only map it between different kernel sub-systems. This means that target applications comprise data storage servers for applications that do not manipulate data in any way, i.e., no data touching operations are performed by the application. Examples of such mechanisms are the *stream* system call [6], the Hi-Tactix system [7], KStreams [8] and the *sendfile* system call (for more references, see [2]).

Besides memory movement, system calls are expensive operations, because each call to the kernel requires two switches. Even though in-kernel data paths remove some of this overhead, many applications still require application level code that makes kernel calls. Relevant approaches to increase performance include batched system calls [9] and event batching [10].

Although these examples show that an extensive amount of work has been performed on copy and system call avoidance techniques, the proposed approaches have usually remained research prototypes for various reasons, e.g., they are implemented in own operating systems (having an impossible task of competing with Unix and Windows), small implementations for testing only, not integrated with the main source tree, etc. Therefore, only some limited support is included in the most used operating systems today like the *sendfile* system call in UNIX, Linux, AIX and *BSD. In the next section, we therefore evaluate the I/O pipeline performance of the new Linux 2.6 kernel.

3 Existing Mechanisms in Linux

Despite all the proposed mechanisms, only a limited support for various streaming applications is provided in commodity operating systems like Linux. The existing solutions for moving data from storage device to network device usually comprise combinations of the *read/write*, *mmap* and *sendfile* system calls. Below, we present the results of our performance tests using combinations of these for **content download** operations (adding no application level information) and **streaming** operations (adding application level RTP header for timing and sequence numbering).

3.1 Test Setup

The experiments were performed using two machines connected by a point-to-point Ethernet connection. The test machine has an Intel 845 chipset, 1.70 GHz Intel Pentium4 CPU, 400 MHz front side bus and 1 GB PC133 SDRAM. The resource usage is measured using the *getrusage* function measuring consumed user and kernel time to transfer 1 GB of data stored using the Reiser file system in Linux 2.6. Below, we have added the user and kernel time values to get the total resource consumption, and each test is performed 10 times to get more reliable results. However, the differences between the tests are small.

3.2 Copy and Switching Performance

Before looking at the disk-network data transfer performance, we first look at the memory copy and system call performance themselves. In figure 1, an overview of the chipset on our test machine is shown (similar to many other Intel chipsets). Transfers between device and memory are typically performed using DMA transfers that move data over the PCI bus, the I/O controller hub, the hub interface, the memory controller hub and the RAM interfaces. A memory copy operation is performed moving data from RAM over the RAM interfaces, the memory controller hub, the system (front side) bus through the CPU and back to another RAM location with the possible side effect of flushing the cache(s). Data is (possibly unnecessarily) transferred several times through shared components reducing the overall system performance.

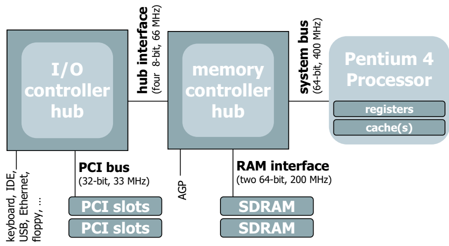
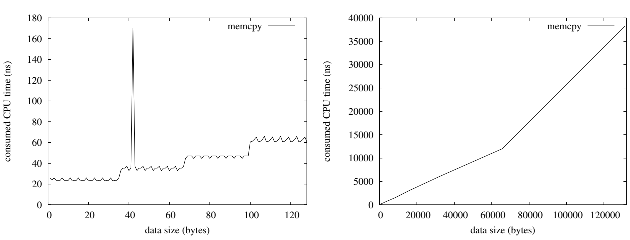


Figure 1. Pentium4 processor and 845 chipset

In [11], memory copy performance was measured on Linux (2.2 and 2.4) and Windows (2000) where the conclusion was that *memcpy* performs well (compared to other copy functions/instructions), and Linux is in most cases faster than Windows depending on data size and used copy instruction. Furthermore, to see the performance on our test machine, we tested *memcpy* using different data sizes. Figure 2a shows that the overhead is slowly growing with the size of the data element, but after reaching a certain size (having cache size effects) the overhead increases more or less linearly with the size (figure 2b)[1].



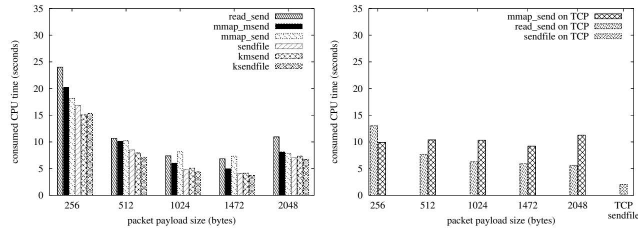
a) b)

Figure 2. User space memory copy speed

With respect to switching contexts, system call overhead (and process context switches) are time consuming on Pentium4 [12]. To get an indication of the system call overhead on our machine, we measured the *getpid* system call, accessing the kernel and only returning the process id. Our experiments show that the average time to access the kernel and return back is approximately 920 nanoseconds for each call.

Copy and system call performance has also been an issue for hardware producers like Intel, who has added new instructions, in particular MMX and SIMD extensions useful for copy operations and *sysenter* and *sysexit* instructions particularly for system calls. For example, using SIMD instructions, the block copy operation speed was improved by up to 149% in the Linux 2.0 kernel, but the reduction in CPU usage was only 2% [13]. Thus, both copy and kernel access performance still are resource consuming and remain possible bottlenecks.

[1]The single high peek in figure 2a at 42 bytes is due to one single test being interrupted by an external event. Additionally, some strange artifacts can be seen when the size of the memory address are not 4-byte aligned with even higher peeks in user space. This is also apparent for larger data sizes and is probably due to different instructions.



a) UDP b) TCP

Figure 3. Content download operations

3.3 Disk-Network Path

The functions used for retrieving data from disk into memory are usually *read* or *mmap*. Data is transfered using DMA from device to memory, and in case of *read*, we require an in-memory copy operation to give the application access to data whereas *mmap* shares data between kernel and user space. To send data, *send* (or similar) can be used where the payload is copied from the user buffer or the page cache depending on whether *read* or *mmap* is used, respectively, to the socket buffer (*sk_buf*). Then, the data is transfered in a DMA operation to the network device. Another approach is to use *sendfile* sending the whole file in one operation, i.e., data is sent directly from a kernel buffer to the communication system using an in-kernel data path. Thus, if gather DMA operations are supported, i.e., needed because the payload and the generated headers are located in different (*sk_buf*) buffers, data can be sent from disk to network without any in-memory copy operations.

3.4 Content Download Experiments

The first test we performed looking at the whole disk-network data path was in a content download scenario. Here, data needs only to be read from disk and sent as soon as possible without application level control. Thus, there is no need to add application level information. To evaluate the performance, we performed several tests using the different data paths and system calls described in section 3.3 and table 1-A using both TCP and UDP. For UDP we also added three enhanced system calls to be able to test a download scenario similar to *sendfile* with TCP.

The results for UDP are shown in figure 3a. We see, as expected, that removal of copy operations and system calls both give performance improvements. Furthermore, in figure 3b, the results using TCP are shown. Again, we see that a quite a lot of resources can be freed using *sendfile* compared to the two other approaches making several system calls and copy operations per data element. Note, however, that it seems that *getrusage* is still not fully implemented for TCP in the 2.6 kernel. Thus, the TCP and UDP experiments are not directly comparable.

From the results, we can see that the existing *sendfile* over TCP performs very well compared to the other tests

A – content download

	co	s	calls to the kernel
read_write	$2n$	$4n$	n ***read*** and n ***write*** calls (TCP will probably gather several smaller elements into one larger MTU-sized packet)
mmap_send	n	2+2n	1 ***mmap*** and n ***send*** calls (TCP will gather several smaller elements into one larger MTU-sized packet)
sendfile (UDP)	0	$2n$	n ***sendfile*** calls
sendfile (TCP)	0	2	1 ***sendfile*** call
mmap_msend†	0	2+2n	1 ***mmap*** and n ***msend*** calls (*msend*† sends data over UDP using the virtual address of a *mmap*'ed file instead of copying the data)
kmsend†	0	2+2n	1 ***kmsend*** call (*kmsend*† combines *mmap* and *msend* (see above) in the kernel until the whole file is sent)
ksendfile†	0	2	1 ***ksendfile*** call (*ksendfile*† performs *sendfile* over UDP in the kernel a la *sendfile* for TCP)

B – RTP streaming

	co	s	calls to the kernel
read_send_rtp	$2n$	$4n$	n ***read*** and n ***send*** calls (RTP header is placed in user buffer in front of payload, i.e., no extra copy operation)
read_writev	$3n$	$4n$	n ***read*** and n ***writev*** calls (RTP header is generated in own buffer, *writev* write data from two buffers)
mmap_send_rtp	$2n$	2+8n	1 ***mmap***, n ***cork***, n ***send***, n ***send*** and n ***uncork*** calls (need one send call for both data and RTP header)
mmap_writev	$2n$	2+2n	1 ***mmap*** and n ***writev*** calls
rtp_sendfile	n	$8n$	n ***cork***, n ***send***, n ***sendfile*** and n ***uncork*** calls

C – enhanced RTP streaming

	co	s	calls to the kernel
mmap_rtpmsend†	n	2+2n	1 ***mmap*** and n ***rtpmsend*** calls (*rtpmsend*† copies RTP headers from user space and adds payload from *mmap*'ed files as payload in the kernel)
mmap_send_msend†	n	2+8n	1 ***mmap***, n ***cork***, n ***send***, n ***msend*** and n ***uncork*** calls (no data copying using *msend*, but the RTP header must be copied from user space)
rtpsendfile†	n	$2n$	n ***rtpsendfile*** calls (*rtpsendfile*† adds the RTP header copy operation to the *sendfile* system call)
krtpsendfile†	0	2	1 ***krtpsendfile*** call (*krtpsendfile*† adds RTP headers to *sendfile* in the kernel)
krtpmsend†	0	2	1 ***krtpmsend*** call (*krtpmsend*† adds RTP headers to the *mmap*/*msend* combination (see above) in the kernel)

co = number of *copy operations*, s = number of *switches* between user and kernel space, n = number of packets, † = new enhanced system call

Table 1. Descriptions of the performed tests

as applications only have to make one single system call to transfer a whole file. Consequently, if no data touching operations, no application level headers or timing support are necessary, *sendfile* seems to be efficiently implemented and achieves a large performance improvement compared to the traditional *read* and *write* system calls, especially when using TCP where only one system call is needed to transfer the whole file.

3.5 Streaming Experiments

Streaming time-dependent data like video to remote clients typically requires adding per-packet headers, such as RTP headers for sequence numbers and timing information. Thus, plain file transfer optimizations are insufficient, because file data must be interleaved with application generated headers, i.e., additional operations must be performed. To evaluate the performance of the existing mechanisms, we performed several tests using the set of data paths and system calls listed in table 1-B. As shown above, the application payload can be transfered both with and without user space buffers, but the RTP header must be copied and interleaved within the kernel. Since TCP may gather several packets into one segment, i.e., the RTP headers will be useless, we have only tested UDP. The results of our tests are shown in figure 4a. Compared to the ftp-like operations in the previous section, we need many system calls and copy operations. For example, compared to the *sendfile* (UDP) and the enhanced *ksendfile* tests in figure 3, there are a 21% and a 29% increase in the measured overhead for the *rtp_sendfile* using Ethernet MTU-sized packets, respectively. This is because we now also need an additional *send* call for the RTP header. Thus, the results indicate that there is a potential for improvements. In the next section, we therefore describe some possible improvements and show that already minor enhancements can achieve large gains in performance.

4 Enhancements for RTP streaming

Looking at the existing mechanisms described and analyzed in the previous section, we are more or less able to remove copy operations (except the small RTP header), but the number of user/kernel boundary crossings is high. We have therefore implemented a couple of other approaches listed in table 1-C. With respect to overhead, *mmap_rtpmsend*, *rtpsendfile*, *krtpmsend* and *krtpsendfile* look promising:

- *mmap_rtpmsend* uses *mmap* to share data between file system buffer cache and the application. Then, it uses the enhanced *rtpmsend* system call to send data copying a user-level generated RTP header and adding the mapped file data using a virtual memory pointer instead of a physical copy. This gives n in-memory data transfers and $1 + n$ system calls. (A further improvement would be to use a virtual memory pointer for the RTP header as well)
- *krtpmsend* uses *mmap* to share data between file system buffer cache and the application and uses the en-

hanced *msend* system call to send data using a virtual memory pointer instead of a physical copy. Then, the RTP header is added in the kernel by a kernel-level RTP engine. This gives no in-memory data transfers and only 1 system call.

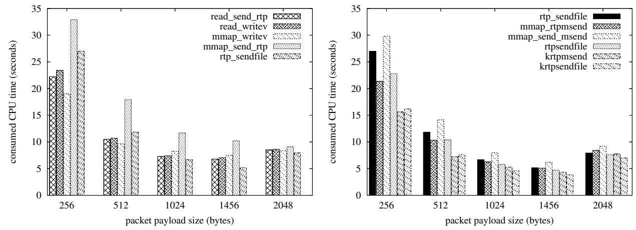
- *rtpsendfile* is a modification of the *sendfile* system call. Instead of having an own call for the RTP header transfer, an additional parameter (a pointer to the buffer holding the header) is added, i.e., the data is copied in the same call and sent as one packet. This gives only n in-memory data transfers and n system calls.
- *krtpsendfile* uses *ksendfile* to transmit a UDP stream in the kernel, in contrast to the standard *sendfile* requiring one system call per packet for UDP. Additionally, the RTP header is added in the kernel having an in-kernel RTP engine. This gives no in-memory data transfers and only 1 system call.

The two first mechanisms are targeted at applications requiring the possibility to touch data in user-space, e.g., parsing or sporadic modifications[2], whereas the last two mechanisms aim at data transfers without application-level data touching. All these enhanced system calls reduce the overhead compared to existing approaches, and to see the real performance gain, we performed the same tests as above. Our results, shown in figure 4b, indicate that simple mechanisms can remove both copy and system call overhead. For example, in the case of streaming using RTP, we see an improvement of about 27% using *krtpsendfile* where a kernel engine generates RTP headers compared to *rtp_sendfile* in the scenario with MTU-sized packets. If we need the same user level control making one call per packet, the *rtpsendfile* enhancement gives at least a 10% improvement compared to existing mechanisms. In another scenario where the application requires data touching operations, the existing mechanism only have small differences. If comparing the results for MTU-sized packets, *read_send_rtp* (already optimized to read data into the same buffer as the generated RTP header) performs best in our tests. However, using a mechanism like *krtpmsend* gives a performance gain of 36% compared to *read_send_rtp*. Similar user level control by making one call per packet is achieved by *mmap_rtpmsend* which gives a 24% gain. Additionally, similar results can in general also be seen for smaller packet sizes (of course with higher overhead due to a larger number of packets), and when the transport level packet exceeds the MTU size, additional fragmentation of the packet introduces additional overhead.

5 Discussion

The enhancements described in this paper to reduce the number of copy operations and system calls mainly address application scenarios where data is streamed to the client

[2]Non-persistant modifications to large parts of the files require a data copy in user space, voiding the use of the proposed mechanisms.



a) Existing mechanisms b) Enhancements

Figure 4. Streaming performance

without any data manipulation at the server side. However, several of the enhanced system calls also show that the application can share a buffer with the kernel and can interleave other information into the stream. Thus, adding support for data touching operations, like checksumming, filtering, etc. without copying, and data modification operations, like encryption, transcoding, etc. with one copy operation, should be trivial. The in-kernel RTP engine also shows that such operations can be performed in the kernel (as kernel stream handlers), reducing copy and system call overhead.

An important issue is whether data copying is still a bottleneck in systems today. The hardware has improved, and one can easily find other possible bottleneck components. However, as shown in section 3.2, data transfers through the CPU are time and resource consuming and have side effects like cache flushes. The overhead increases approximately linearly with the amount of data, and as the gap between memory and CPU speeds increases, so does the problem. Additionally, in figure 5 (note that the y-axis starts at 0.5), we show the performance of the different RTP streaming mechanisms relative to *read_writev*, i.e., a straight forward approach reading data into an application buffer, generating the RTP header and writing the two buffers to the kernel using the vector write operation. Looking for example at MTU-sized packets, we see that a lot of resources can be freed for other tasks. We can also see that less intuitive but more efficient solutions than *read_writev* that do not require kernel changes exist, for example using *sendfile* combined with a *send* for the RTP header (*rtp_sendfile*). However, the best enhanced mechanism, *krtpsendfile*, removes all copy operations and makes only one access to the kernel compared

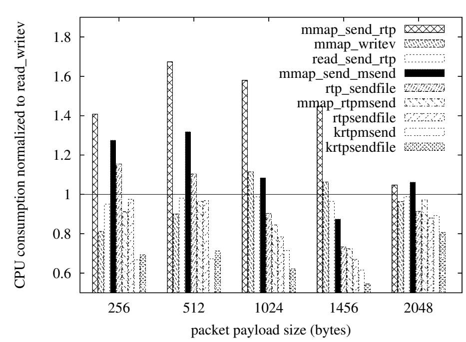


Figure 5. Relative performance to *read_writev*

to *rtp_sendfile* which requires several of both (see table 1). With respect to consumed processor time, we achieve an average reduction of 27% using *krtpsendfile*. Recalculating this into (theoretical) throughput, *rtp_sendfile* and *krtpsendfile* can achieve 1.55 Gbps and 2.12 Gbps, respectively. Assuming a high-end 3.60 GHz CPU like Pentium4 660 and an 800 MHz front side bus, the respective numbers should be approximately doubled. These and higher rates are also achievable for network cards (e.g., Force10 Network's E-Series), PCI express busses and storage systems (e.g., using several Seagate Cheetah X15.4 in a RAID). Thus, the transfer and processing overheads are still potential bottlenecks, and the existing mechanisms should be improved.

Now, having concluded that data transfers and system calls still are possible bottlenecks and having looked at possible enhancements, let us look at what a general purpose operating system like Linux miss. Usually, the commodity operating systems aim at a generality, and new system calls are not frequently added. Thus, specialized mechanisms like *krtpsendfile* and *krtpmsend* having application specific, kernel-level RTP-engines, will hardly ever be integrated into the main source tree and will have to live as patches for interested parties like streaming providers, e.g., like the Red Hat Content Accelerator (*tux*) for web services. However, support for adding application level information (like RTP headers) to stored data will be of increasing importance in the future as streaming services really explode. Simple enhancements like *mmap_rtpmsend* and *rtpsendfile* might be general, performance improving mechanisms that could be of interest in scenarios where the application does or does not touch the data, respectively.

6 Conclusions

In this paper, we have shown that (streaming) applications still pay a high (unnecessary) performance penalty in terms of data copy operations and system calls if those applications require packetization such as addition of RTP headers. We have therefore implemented several enhancements to the Linux kernel, and evaluated both existing and the new mechanisms. Our results indicate that data transfers still are potential bottlenecks, and simple mechanisms can remove both copy and system call overhead if a gather DMA operation is supported. In the case of a simple content download scenario, the existing *sendfile* is by far the most efficient mechanism, but in the case of streaming using RTP, we see an improvement of at least 27% over the existing methods using MTU-sized packets and the *krtpsendfile* system call with a kernel engine generating RTP headers. Thus, using mechanisms for more efficient resource usage, like removing copy operations and avoiding unnecessary system calls, can greatly improve a node's performance. Such enhancements free resources like memory, CPU cycles, bus cycles, etc. which now can be utilized by other applications or providing support for more concurrent streams.

Currently, we also have other kernel activities on-going, and we hope to be able to integrate our subcomponents. We will also modify the KOMSSYS video server to use the proposed mechanisms and perform more extensive tests including a workload experiment looking at the maximum number of concurrent clients able to achieve a timely video playout. Finally, we will optimize our implementation, because most of the enhancements are implemented as proof-of-concept removing copy operations and system calls. We have made no effort in optimizing the code, so the implementations have large potential for improvement, e.g., moving the send-loop from the system call layer to the page cache for the *krtpsendfile* which will remove several file lookups and function calls (as for the existing *sendfile*).

References

[1] Vivek S. Pai, Peter Druschel, and Willy Zwaenepoel. Io-lite: a unified i/o buffering and caching system. *ACM Transactions on Computer Systems*, 18(1):37–66, February 2000.

[2] Pål Halvorsen. *Improving I/O Performance of Multimedia Servers*. PhD thesis, Department of Informatics, University of Oslo, Oslo, Norway, August 2001.

[3] Daniel G. Bobrow, Jerry D. Burchfiel, Daniel L. Murphy, and Raymond S.Tomlinson Bolt Beranek. Tenex, a paged time sharing system for the pdp-10. *Communications of the ACM*, 15(3):135–143, March 1972.

[4] Eric W. Anderson. *Container Shipping: A Uniform Interface for Fast, Efficient, High-Bandwidth I/O*. PhD thesis, Computer Science and Engineering Department, University of California, San Diego, CA, USA, July 1995.

[5] Charles D. Cranor and Gurudatta M. Parulkar. The UVM virtual memory system. In *Proceedings of the USENIX Annual Technical Conference*, pages 117–130, Monterey, CA, USA, June 1999.

[6] Frank W. Miller and Satish K. Tripathi. An integrated input/output system for kernel data streaming. In *Proceedings of SPIE/ACM Conference on Multimedia Computing and Networking (MMCN)*, pages 57–68, San Jose, CA, USA, January 1998.

[7] Damien Le Moal, Tadashi Takeuchi, and Tadaaki Bandoh. Cost-effective streaming server implementation using hi-tactix. In *Proceedings of the ACM International Multimedia Conference (ACM MM)*, pages 382–391, Juan-les-Pins, France, December 2002.

[8] Jiantao Kong and Karsten Schwan. Kstreams: Kernel support for efficient end-to-end data streaming. Technical Report GIT-CERCS-04-04, College of Computing, Georgia Institute of Technology, Atlanta, GA, USA, 2004.

[9] Charles Coffing. An x86 protected mode virtual machine monitor for the mit exokernel. Master's thesis, Paralell & Distributed Operating System Group, MIT, Cambridge, MA, USA, May 1999.

[10] Christian Poellabauer, Karsten Schwan, Richard West, Ivan Ganev, Neil Bright, and Gregory Losik. Flexible user/kernel communication for real-time applications in elinux. In *Proceedings of the Workshop on Real Time Operating Systems and Applications and Second Real Time Linux Workshop*, Orlando, FL, USA, November 2000.

[11] Edward Bradford. Runtime: Block memory copy (part 2) – high-performance programming techniques on linux and windows. http://www-106.ibm.com/developerworks/library/l-rt3/, July 1999.

[12] Gregory McGarry. Benchmark comparison of netbsd 2.0 and freebsd 5.3. http://www.feyrer.de/NetBSD/gmcgarry/, January 2005.

[13] Intel Corporation. Block copy using PentiumIII streaming SIMD extensions (revision 1.9). ftp://download.intel.com/design/servers/-softdev/copy.pdf, 1999.

Application of Heuristic MMKP in Admission Control and QoS Adaptation for Distributed Video on Demand Service

Md. Shamsul Alam S.M.Kamrul Hasan A.S.M. Sohail Mahmudul Hasan Boshir Ahmed
Department of Computer Science and Engineering
Rajshahi University of Engineering & Technology
Rajshahi-6204, Bangladesh
{shamsulbd, khrajeeb, sohail_ruet, hsagor, boshir_bd}@yahoo.com

Abstract

Allocation and reservation of resources, such as CPU cycles and I/O bandwidth of multimedia servers and link bandwidth in the network, is essential to ensure Quality of Service (QoS) for multimedia services delivered over the Internet. In this paper, we have proposed a new semi-distributed architecture for admission control and QoS adaptation of multimedia sessions to maximize revenue from multimedia services for Distributed Video on Demand System (DVoDS). We have introduced the mapping of Utility Model - Distributed (UM-D) by semi-distributed controller to the Multidimensional Multiple-choice Knapsack Problem (MMKP), a variant of the classical 0-1 Knapsack Problem. An exact solution of MMKP, an NP-hard problem, is not applicable for the online admission control problem in the VoD System. Therefore we have applied heuristic, I-HEU for solving the MMKP for online real-time admission control and QoS adaptation. We have applied the admission control strategy described in the UM-D to the set of Media Server Farms providing streaming videos to users. The performance of semi-distributed architecture applied in a simulated environment over a set of Media Server Farm has been discussed detail using the experimental outcome.

1. Introduction

Recent years have witnessed the use of digital video and audio in several important applications that affect various aspects of our daily life. These applications include distance learning, digital libraries, Movie-on-Demand (MoD), electronic commerce, etc. Some applications, such as video conferencing or streaming audio, video on demand, produce network traffic, which require a guaranteed level of Quality-of-Service (QoS) to work properly. These QoS requirements may be in terms of a minimum bandwidth, bounded end-to-end delays, or maximum packet loss rate suffered by a flow. To support such flows and to allocate and maintain their finite network resources to uphold their guarantees, some form of admission control is important to accept profitable flows and to reject new traffic flows that would cause the controller to violate its promises. In this paper we have proposed a semi-distributed model of VoD system and semi-distributed controller architecture to do admission control and QoS adaptation for the set of Media Server Farm. We have proposed the mapping of utility model-distributed (UM-D) to the MMKP for the semi-distributed admission controller. Finally, to analyze the performance, we have presented simulation of admission controllers over a set of Media Server Farms.

2. Related Work

Guaranteed transmission of voice and video is a challenge of the researchers in Internet computing field. There has been lot of interesting works in recent years on reservation-based management of resources like CPU cycles and bandwidth carrying the video and voice [1, 2, 3]. Transmission of video requires guaranteed Quality of Service (QoS), which is defined by the amount of bandwidth, latency and jitter bound; otherwise the customers will not pay for the services. Delivery of multimedia streams with absolute guaranteed QoS from a single multimedia server has been proposed by Khan [4]. Hua[1] describes video delivery using multicast, streaming strategies with application layer multicast, and proxy caching techniques. To do admission control and QoS adaptation in a set of Media Server Farms, centralized broker architecture has been proposed by Akbar [5]. The load balancing problems of centralized scheme lead us to the development of a fully distributed scheme for MM servers with more scalability and fault tolerance [7]. But, fully distributed architecture faces message passing complexity and time requirement overhead for specific session. Mundur [8] proposed a global request handling and admission control strategies based on limited redirection of blocked requests to other resources for distributed VoD system. Redirect request handling polices require higher implementation overhead and extra connection setup time. The Utility Model- Distributed (UM-D) [4,6] provides a sound strategy for selecting an optimal set of sessions, with specified QoS levels and a set of servers to provide each of MM video streams. The goal is to guaranteeing the agreed levels of QoS, facing resource constraints of the servers and the network. The Utility Model is mapped to the MMKP, whose exact solution is NP hard. The proposed heuristic algorithm [6] is suitable enough for finding a set of QoS levels for the multimedia sessions quickly to allow admission and adaptation decisions in real time.

3. Preliminaries

3.1 The Multidimensional Multiple Choice Knapsack Problem(MMKP)

The MMKP is a variant of the KP. Let there be *n groups of items*. Group i has l_i items. Each item of the group has a value and its required quantities of each of the m resources. The objective of MMKP is to pick up exactly one item from each group, for maximum total value of the collected items, subject to m resource constraints of the knapsack [6].

3.2 Utility Model - Distributed (UM-D)

The Utility Model - Distributed provides a strategy for selecting an optimal set of sessions (with particular QoS levels and selection of servers to provide multimedia stream components) from the submitted requests for multimedia sessions. The goal is to maximize the revenues earned from the selected sessions, subject to the resource constraints imposed by the servers and networks of the DVoDS. The UM-D mainly describes the admission control and QoS adaptation strategy for the multimedia sessions in mathematical notation [4,5].

3.3 Video on demand service

A Video on Demand (VoD) system consists of a video server with a video archive and a number of client machines connected via a local area network. Users use client software to make a request for their desired video. In response to a service request, the server delivers the requested video to the user. The VoD service is replicated in order to achieve high availability and fault tolerance. Each video file is held by a subset of the servers. The service is provided by a number of servers having the capability of dynamic group formation. The service from the servers to carry multimedia video streams for all customers is provided over a local distribution network, such as ATM LAN, HFC, or xDSL. It is assumed that there are sufficient network resources at the local distribution network to deliver videos to the customers and that there is no resource contention on the local distribution network.

4. Distributed admission control architecture

4.1 Broker architecture

Akbar [5] presented centralized broker architecture. This architecture utilizes centralized computing performed by a broker. The broker is interposed between the clients and servers of the system. This architecture faces load-balancing problem with the increase of users.

4.2 Semi-distributed architecture

In this approach, each Media Server Farm (MSF) is connected with a Local Controller (LC) and all the LCs are interconnected through a Central Controller (CC) for negotiation among them. Only the LC of a MSF has the authority to allocate resources of the servers for providing VoD services to the users. Each LC keeps the record of available resources of the corresponding media servers. Users are required to communicate with (one of) the local admission controllers of the DVoDS to submit their requests and to receive admission or rejection decisions made by the controllers. The semi-distributed admission control and QoS adaptation architecture is shown in figure1.

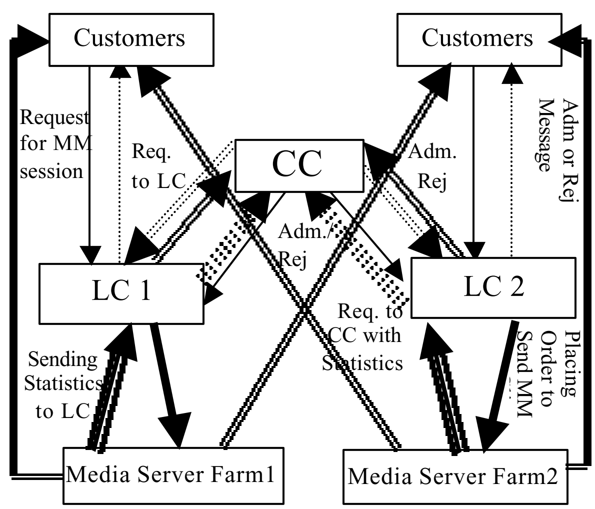


Figure 1: The semi-distributed admission control and QoS adaptation architecture

Users send their request for MM streams to the LC and get an admission if the local server has capacity of giving MM streams. If the local server has no capacity to give MM streams, then it forwards the request to the CC, which checks the available resources of other LCs automatically and finally send the admission request to the possibly appropriate LC or rejection message to the requesting user. The underlying network must be able to accommodate the messages required for the admission control and QoS adaptation processes.

5. Mapping of the UM-D by semi-distributed architecture to the MMKP

Admission control and QoS adaptation by the semi-distributed architecture can be mapped to the MMKP. Each server in the DVoDS represents one knapsack with multidimensional resources such as CPU cycles, I/O and Memory. Here, we have ignored the link bandwidths for the simplicity of mapping. Knapsack Problem specifies resource management. Our policy is to do admission control by allocating resources. That is why Knapsack Problem is suitable to represent the admission control problem.

5.1 Mapping of sessions to the groups

Each multimedia session request with a set of QoS levels submitted to the controller, represents a group in the MMKP.

5.2 Mapping of the QoS levels of a session to the items of a group

A particular level of QoS $\vec{q}_{ij}$ can be provided by one or more servers, because the components of the multimedia stream can be partitioned and replicated in multiple servers. We can find all the combinations of servers that can serve a particular level of QoS of a particular requested multimedia session. Let there be M servers. Now this level of QoS can be provided by M combinations of servers These combinations might have different utilities. These are indistinguishable from the user's point of view, but are very different from the DVoDS controller's point of view. Thus one QoS level is transformed to multiple options. Each of these options is defined as QoS with respect to Controllerr (QoS-C) and represents an item of a group in the MMKP.

5.3 Mapping of server resources to the resources of the knapsack

If there are m_c resource dimensions in each of M multimedia servers, then we can think of a merged server with M m_c resources; it is in effect the union of the physical servers, and its resources belong to the cross-product space (server, resource). In this way we can map multiple servers to one merged server and hence to one knapsack. Now, our problem is to find a QoS-C level for each session for maximum utility of the DVoDS, subject to resource constraints of the merged server, which is treated as an MMKP.

6. Admission control and QoS adaptation methodology

New sessions are collected into a batch over a time interval. We have used I-HEU, the incremental heuristic solution of the MMKP proposed in [6] for admission control and QoS adaptation. A batch of multimedia sessions is submitted to the controller which finds out the QoS-C levels. For the new sessions, a dummy null level is added to the QoS-C profile of each session. This QoS level gives null revenue with null resource consumption. I-HEU is applied to the new sessions as well as to the already admitted one, once in every batch. I-HEU finds the new set of QoS-C levels for each session by upgrading, by downgrading, or without changing the QoS-C levels of the sessions in order to maximize the earned revenue from the sessions. If the QoS-C level of any new session in the new batch remains null after applying I-HEU, then it is rejected. The other sessions are admitted at their non-null QoS-C levels. The admitted sessions enjoy the assigned QoS-C level at least for the next epoch.

When any session leaves, some resources are released from the multimedia server system. The computation of I-HEU for the next batch will get the opportunity of these released resources to upgrade some sessions or to admit more sessions.

When some resources of the system decreases, for example if the communication bandwidth decreases due to the failure of any communication link or a server goes down, then QoS adaptation is done as soon as the fault is detected. It is done in three steps:

Step 1: All the admitted sessions are downgraded to the lowest QoS level (not null). If I-HEU gives a feasible solution then adaptation is complete. Otherwise execute step 2.

Step 2: All the admitted sessions are downgraded to the null QoS level. Now I-HEU will admit some sessions and some may be rejected. If any session is rejected then execute step 3.

Step 3: All the sessions admitted in step 2 are downgraded to the lowest level of QoS. Now some resource will be released and I-HEU is applied to the rejected sessions from step 2. This will allow some sessions to be admitted and the rest will be rejected. These rejected sessions will not be considered further. The admitted sessions in step 2 and step 3 will be upgraded again by I-HEU to do final adaptation.

7. Computational complexity of the controller

The complexity of the controller depends on the size of the system, the number of users in the system and the algorithm used to solve the MMKP. We have presented the complexity analysis of the broker initialized here to run the simulation over a set of Media Server Farms.

The total number of users in this set of Media Server Farms with M servers is approximately M n_k, where $n_k =$

estimated number of users enjoying Silver QoS from a server at full load. There are M video servers. Each server has three resources, CPU cycles, Memory and I/O BW. So, the total number of resources = $3M$.

Each video stream is replicated in more than one server. The number of replications can be expressed by $M \quad 2/10$. On the average, there are 2 QoS levels in each session. So, the number of QoS-B levels for a session is approximately M/5. The MMKP solved by the broker has $n \quad M \quad n_k$ groups, l=M/5 items in each group and $m = 3M$ resource dimensions. Using the complexity analysis of the heuristics of the MMKP, we get the following worst-case complexity of the controller using I-HEU,

$$O \quad \frac{M^2 n_k^2 M^2 3M}{25} \qquad O\,M^5 \quad .$$

8. Simulation of the admission controller

We have simulated the admission controllers of the set of Media Server Farms for different distributed control architectures using greedy and heuristic approach. The performance data has been collected by varying the number of users and the number of video servers. The following tables present the parameters of the set of Media Server Farms required for the simulation.

Table 1: Different simulation parameters

Parameter	Value(s) for the Simulation								
Maximum length of movie	3 hours								
Total Number of movies in MSF	10								
Number of video servers in MSF	5	10	15	20	25	30	35	40	45
Number of replications of video	3	3	3	4	5	6	7	8	9
Number of movie copies per server	6	3	2	2	2	2	2	2	2
Repetitions of the simulation experiment	5								

The servers are initialized with CPU cycles, memory and I/O bandwidth capacity, where all the parameters are randomized by uniform distribution *U (0.95, 1.05)*, to simulate the fluctuation of the available resources in different servers. The users can enjoy the movie in three QoS levels (Gold, Silver, Bronze), where each QoS level resource requirement has been defined with randomization using *U (0.75, 1.25)*.

Table 2: Initialization of server resources in the media server farms

Type of resource	Server resource value	Initialization function for a video server	Assumptions
CPU	400 MHz dual processor	100 X U(0.95, 1.05)	Total 800 MHz is equivalent to 100 cycles.
RAM	256 MB	120 X U(0.95, 1.05)	136 MB is used by the O/S.
Bandwidth	640 Mbps	600 X U(0.95, 1.05)	40 Mbps is reserved for the system.

Table 3: Different QoS levels supported by the DVoDS

QoS levels	Average I/O BW req.	Average CPU cycles req.	Average Memory req.	Average offered price by the user	Average cost of providing media
QoS 1 (Bronze)	1.5 Mbps	0.25%	0.3 MB	\$1.0	\$0.75
QoS 2 (Silver)	3.0 Mbps	0.50 %	0.6 MB	\$3.0	\$2.25
QoS 3 (Gold)	4.5 Mbps	0.75%	0.9 MB	\$5.0	\$3.75

Table 4: Initialization of resources requirements for differnt QoS levels

Parameter	Initialization function
I/O bandwidth requirement for the kth QoS level	1.5 X k X U(0.75, 1.25)
Memory requirement for the kth QoS level	0.3 X k X U(0.75, 1.25)
CPU cycles for the kth QoS level	0.25 X k X U(0.75, 1.25)
Offered price by a user for the kth QoS level	(2k+1) X U(0.75, 1.25)
Cost of providing kth QoS level of a movie	(2k+1) X U(0.55, 0.95)

We have coded the simulation of the LC & CC using the Java programming language (JDK ver 1.5). LC generates a big batch of user session request initially, having double size of Media Server Farm's server resource capacity, to get admission for creating contention of resources in the resided servers. We have discarded 10 percent of accepted sessions to create space for future sessions, then generate 15 percent new sessions as a small batch. The total simulation ends when all LCs have generated 100 batch. We have calculated the Revenue earned from the users and the time required by LC to do admission control and QoS adaptation. The summation of the revenues of the sessions and average time requirement by each session is the measurement criteria of the distributed admission controller's performance in our simulation. The simulation was implemented on a 1.7 GHz Pentium IV machine plugged in a LAN and was running under Windows XP having a physical memory of 128MB RAM.

9. Experimental Results

Figure2 and figure3 depict that our new semi-distributed architecture requires much less acceptation and rejection time than centralized broker architecture in both heuristic I-HEU and greedy approach. This is because the total workload is distributed among the LCs in semi-distributed architecture and each LC fetches less traffic than one centralized server with one centralized controller in the broker architecture.

In both architectures, the greedy approach seems to be good than I-HEU with respect to response time as it takes less acceptation and rejection time than I-HEU. This is because, I-HEU is applied to the new sessions as well as to the already admitted ones, once in every batch. I-HEU finds the new set of QoS-C levels for each session by upgrading, by downgrading, or without changing the QoS-C levels of the sessions, in order to maximize the earned revenue from the sessions. For this reason I-HEU requires higher time than greedy.

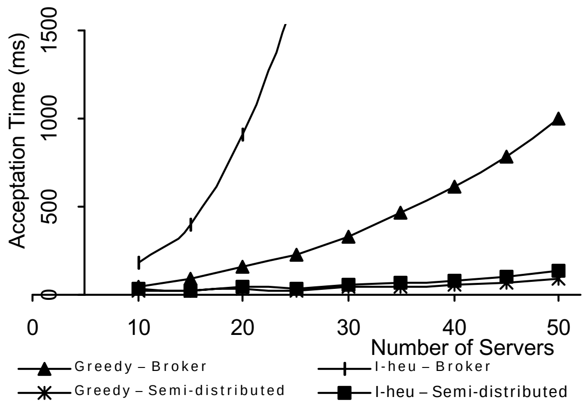


Figure 2: Average acceptation time for each session in a dmission controller (AC).

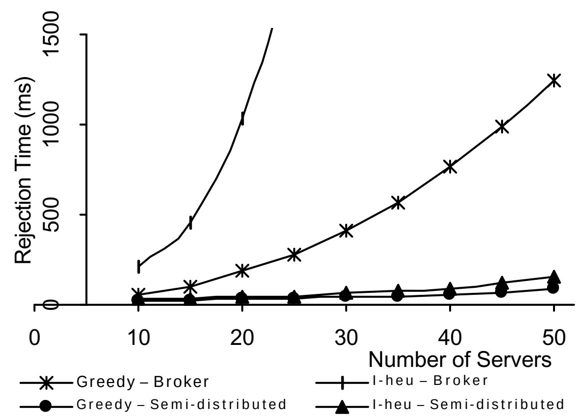


Figure 3: Average rejection time for each session in AC

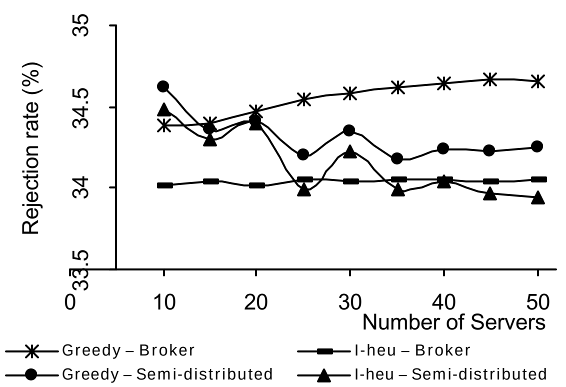


Figure 4: Rejection rate for different number of servers in AC

On the contrary, in figure 4, we see that greedy approach has much higher rejection rate than I-HEU and in figure 5, greedy approach has much less revenue than I-HEU. Our heuristic determines the operating QoS of the admitted session after analyzing all the sessions (admitted and requested) in the system. That is why it gives better results than greedy method, which is based on the FCFS rule and does not search extensively to achieve better utilization of resources. Our main goal is to maximize revenue by admitting more profitable sessions. So, I-HEU gives better result by achieving our objective.

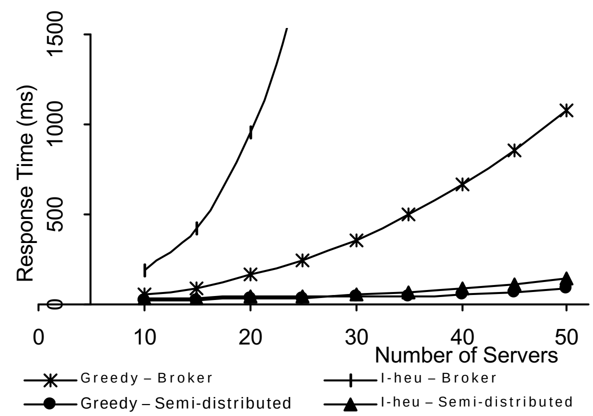


Figure 5: The overall response time for different controller architectures

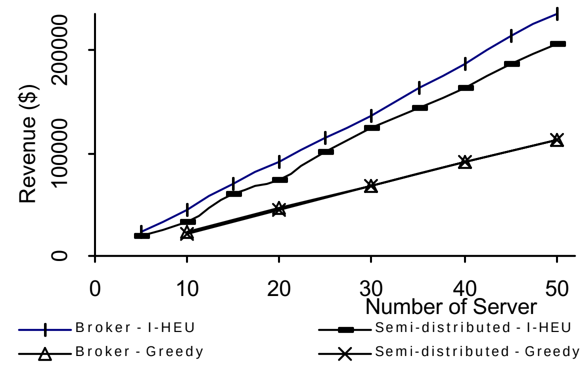


Figure 6: Total revenue earned for different number of servers in AC

The rejection rate of semi-distributed architecture is much less than broker architecture. In both architectures, I-HEU gives better result than greedy approach. The rejection rate of semi-distributed architecture using I-HEU is initially high, but decreases quickly with some fluctuations as the number of server increases and the rejection rate is lowest after 40 servers shown in figure 4. This is because as the number of server increases, there is less resources contention in the system.

Figure 5 illustrates the over all response time of different admission control architectures using different admission control algorithms. Semi-distributed architecture requires fewer overall response time than broker architecture in both approaches. For all admission control architectures, the response time increases if the number of server increases, due to extra search among larger number of servers. So, for a large VoD system our proposed semi-distributed architecture is better and efficient as it provides better fault tolerant.

From figure 6, we observed that the earned revenue for all the architectures increases as the number of server increases. The revenue increases as we add more servers in the system and if there are more users coming proportionally in the system as it gets more users in a batch to select a profitable one. For small number of servers, the controller may select a non-profitable user for admission and may be unable to take advantage of a better session later as resources are reserved by the non-profitable user. On the other hand the larger batch size defers the admission of sessions and the controller may be deprived of earning more money. Sometimes a controller with smaller batch size helps to admit sessions early as some sessions have already been left. Revenue earned is greater when we used I-HEU for both architectures. The semi-distributed architecture has slightly less revenue than broker. But the semi-distributed architecture is inherently more scalable and better fault tolerant which are the key issues for multimedia services.

10. Conclusion

In this paper, we have analyzed the performance of different distributed admission controller architecture using different admission control algorithm. We have presented a new model for admission control and QoS adaptation architecture which addresses the issues of optimal admission of user requested sessions, server failures and scalability that are the key features of video on demand service. The server architecture of the semi-distributed VoD system is analogous to that of a web server in the Internet. The users get access to an intermediate web server and then the page referred to another web server. Our model approximately fits this web server architecture. Experimental results show that the distributed VoD system requires a heuristic admission control algorithm like I-HEU. The semi-distributed VoD system also demonstrates server fault tolerance. The completely distributed computation of I-HEU will become necessary as the load and hence the computational time requirement increases with the increase of users and number of servers. Distribution of controller functionality (admission control and QoS adaptation) leads the system to be more fault tolerant, as the system can run even if one of the admission controllers fails. The concepts demonstrated in this paper may be exploited to construct a highly available distributed VoD service.

11. References

[1] K. A. Hua, M. A. Tantaoui, and W. Tavanapong, “Video delivery technologies for large-scale deployment of multimedia applications,” IEEE Trans. on Multimedia, vol. 92, no. 9, pp. 1439-1451, Sep. 2004.

[2] T. T. Do, K. A. Hua, and M. A. Tantaoui, “P2VoD: Providing fault tolerant video-on-demand streaming in peer-to-peer environment, ” in Proc. IEEE International Conference on Communications (ICC 2004), Paris, France, June 20-24, 2004.

[3] G. Kola, T. Kosar, and M. Livny, “A fully automated fault-tolerant system for distributed video processing and off-site replication,” in Proc. the 14th International Workshop on Network and Operating Systems Support for Digital Audio and Video, Cork, Ireland, pp. 122-126, June 16-18, 2004.

[4] S. Khan, K. F. Li, and E. G. Manning, “The utility model for adaptive multimedia systems,” in Proc. International Conference on Multimedia Modelling, Singapore, pp. 111-126, Nov. 17-20, 1997.

[5] M. M. Akbar, E. G. Manning, and G. C. Shoja, “Admission control and QoS adaptation in distributed multimedia server system,” ITCom 2001, Denver, USA, Aug. 2001.

[6] M M Akbar, E. G. Manning, G. C. Shaja, S. Khan, Heuristic Solutions for the Multiple-Choice Multi-Dimension Knapsack Problem., in Proc. of International Conference on Computational Science, May 2001, San Francis co, USA.

[7] M. S. Alam,and M. W. Haque, “A Fully Distributed Architecture of Video On Demand Services,” 3rd International Conference on Electrical, Electronics and Computer Engineering, ICEECE-2003, Dhaka , Bangladesh, December,2003.

[8] P. Mundur, R. Simon, and A. K. Sood, “End-to-end analysis of distributed video-on-demand systems,” IEEE Trans. on Multimedia, vol. 6, no. 1, pp. 129-141, Feb. 2004.

[9] T. Kameda and S. Wu, “A loss less VoD broadcasting scheme for VBR video using available channel bandwidths,” School of Computing Science, Simon Fraser University, Technical Report 2003-09, Aug. 2003.

Task 3:

On the event of the transmit timer going off

Send all frames to position nodes.

Task 4:

Do any necessary housekeeping functions.

ATC Position Node - p code.

Task 1:

Wait for mark pulse to arrive

From the local node number calculate time of transmission window

Set transmission timer for window.

Task 2:

On the event of the transmission timer going off

Send frame to audio mixer node

Task 3:

Perform any necessary housekeeping functions.**ATC**

It is also possible at this point to describe the automatic fail-over system the system can use. Essentially in the network there will be a number of nodes that listen passively for one of the active nodes to fail, and then take over the responsibilities for that node.

ATC Backup Master Node - p code

Task 1:

Wait for mark pulse to arrive

Re-set watchdog timer for slightly longer than mark period.

Task 2:

On the event of the watchdog timer going off

Restart node as master node.

From examining this design we can see that each node will be allowed to meet the real time constraints that have been laid out for it. The primary challenge for this type of system is to ensure that no two stations transmit at the same time and each station gets an opportunity to transmit; the design presented here does. It has also been seen that the broadcast nature of Ethernet can be an advantage for the purposes of synchronization and redundancy. This design requires that at least two different types of nodes be used in the network. The master node is responsible for providing the synchronization or "mark" pulse to the network. The position nodes use the reception of the "mark" pulse to synchronize the relative times between them and to select unique transmission windows. The tasks required for each type of node to implement the network protocol have been outlined. Each type of node has can be backed up by listening for the failure of a running node. The safety critical aspects of the system have been addressed by outlining the design for backup nodes as well.

CONCLUSIONS

We have successfully designed and implemented a protocol with good real time properties. The protocol developed here is tunable; it is possible to change the parameters of the protocol and trade one positive aspect for another. For example it is possible by increasing the latency in the protocol to increase the data bandwidth available to each node. We can also increase the latency, or decrease the bandwidth to increase the number of permitted nodes on the network.

A simple robust design can support the demands placed upon it. A fundamental characteristic of good design is that it should be no more complicated than necessary. A complex design is much more likely to fail simply by virtue of the fact that there will be more parts to fail. The protocol detailed here is an example of a design that is sufficiently complex to solve the task laid out for it and does not suffer from features that could interfere with its operation.

This work not only develops a theoretical model of a real time network system it provides an example implementation as well. We measured real performance compared to theoretical values for the particular network hardware that was used as well as confirmed the implementation of many aspects of the hardware.

We have created a simple, well designed, largely media independent network protocol that meets real-time requirements functions on cost-effective hardware.

REFERENCES

1. Teledisc Corporation, "Does Latency Matter", December 1997.
2. Andrew S. Tanenbaum, "Computer Networks". Prentice Hall Inc. New Jersey.
3. G. Kramer, B. Mukherjee, and G. Pesavento, “Interleaved Polling with Adaptive Cycle Time (IPACT): A Dynamic Bandwidth Distribution Scheme in an Optical Access Network,” Photonic Network Comm., vol. 4, no. 1,2002, pp. 89-107.
4. Su-il Choi and Jae-doo Huh, Dynamic Bandwidth Allocation Algorithm for Multimedia Services over Ethernet PONs, ETRI Journal, Volume 24, Number 6, Dec. 2002
5. Philip Irey, Robert Harrison and David Marlow, "Techniques for LAN Performance Analysis in a Real Time Environment", Journal of Real-Time Systems, vol 14, pp. 21-44.
6. Motorola - Consumer Systems Group, "Product Brief MFF5307 Integrated Coldfire Microprocessor," Motorola Document #MCF5307/D. 1997.

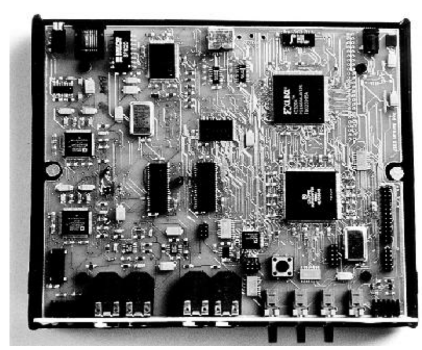


Figure 5: Real Time Network Platform

Extensions of Ethernet for Multimedia Transmission

Donald Molaro
Sr. Software Engineer
Storage Architecture Group (Research)
Hitachi Global Storage Technoligies
Almaden Research Center
650 Harry Road, San Jose, CA

J. R. Parker
Digital Media Laboratory
Department of Computer Science
University of Calgary
1 403 220 6784
parker@cpsc.ucalgary.ca

ABSTRACT

When using a standard Ethernet protocol for the transmission of video and audio, the hard real time requirements of such transmission must be sacrificed. The standard Ethernet protocol does not allow for the guaranteed delivery of media packets on a particular schedule, and so there may be a significant and unpredictable delay that can be seen in the display of the result at the destination. There are ways to adapt the standard Ethernet protocol to allow for real time multimedia transmission and receipt.

Keywords

Multimedia, ethernet, video transmission, networked audio

INTRODUCTION

The world happens in real-time. Computer systems that deal with real world objects, such as computers found in automobiles and consumer electronics, must be concerned with the constraints that time imposes.

However, computers are now being turned into telephones, televisions, radios and other devices that have traditionally not been computer based. Many of these systems deal with streams of data that have the notion of time ingrained into them. Applications that are typical for these systems operate on data has the property that it must arrive at the right time and be intact, the transmitted data must arrive within a specific interval and it must arrive without errors it is time-critical, such as video and audio data. These devices have, until recently, exclusively used analog electronics, which is by its very nature time and order sensitive. Signal loss in these applications results a graceful loss in communications. By comparison the loss of data in a digital stream can result in garbled, incorrect calculations or complete failure of the data transmission.

There is another issue: traditionally, systems that deal with real-time aspects of computers have been limited to either single processor systems or closely coupled multiprocessor systems. Normally, computer network protocols are only concerned with managing the available bandwidth on the medium, and not with promising the timely arrival of video packets. Traditional computer networking protocols have dealt mostly with 'burst" data, and deal poorly with continuous data streams. Indeed, the commonly used networks such as ATM are not so very much concerned with providing the maximum amount of bandwidth but rather providing a consistent quality of service to nodes on the network

We are going to discuss how to network embedded realtime systems for high performance applications, high enough to be used for multimedia applications in local networks. In particular, this paper describes a LAN protocol for real-time distributed systems. This protocol provides guaranteed bandwidth, maximum latency and reliable delivery of data. It takes advantage of current networking protocols and hardware to provide an inexpensive solution that is suitable for embedded systems. Experiments compare actual performance with theoretical predictions, and the results show that this is a useful protocol for systems with real-time data over LANs, including applications ranging from Air Traffic Control, factory automation, data collection in laboratories, and intra-ship communications.

There are several goals of this work. First, we need to devise a protocol with precise, predictable and constant values for network bandwidth, latency, reliability and robustness, which is therefore suitable for implementing distributed hard real time systems. Moreover the design will be as simple as it can be while meeting the requirements as defined. The cost of deploying the functioning system will be a consideration in this project, and the design will reuse as much existing, off the shelf, hardware and software as possible in-order to minimize development time and cost.

NETWORKING REQUIREMENTS

The protocol is to support time synchronized distributed real-time multimedia applications, and multimedia applications contribute a large steady stream of data to the network. Delays in transmission cause noticeable problems; for example, when sending audio across the network, the loss of data result in "clicks" and "pops" being introduced into the playback. These types of artifacts result in unacceptable audio quality and in the case of safety critical applications may be disastrous. The normal burst model used in traditional network analysis is not sufficient here. In interactive media such as in a telephone or video phone the end to end latency in the system must be kept very small and a large buffering of data is not an acceptable option.

Another aspect of media is that each station must gain sufficient access to the network or risk losing data, or falling out of sync with the remote station. As each node in the network collects data it must transmit that data to the receiving node. Nodes in a interactive real-time network must have contiguous and sufficient bandwidth on the network so that they can transmit the required data when they need to.

Networking protocol design alternatives

The process of designing a protocol is to examine the competing paradigms to determine which the requirements of the system fit into, or most closely fit into. The basic families of computer networks are collision detection multiple access (CDMA), frequency division multiple access (FDMA), Time division multiple access (TDMA) and Star networks [Tanenbaum:86].

CDMA networks are common. Ethernet is probably the most commonly used local area networking technology; here, each of the computers shares a common medium and transmits whenever it has data. Specialized circuitry in the network hardware detect when two stations have transmitted at the same time. These systems have the merits of being simple and inexpensive.

FDMA is a system in which each of the nodes in the network is assigned a particular part of the frequency spectrum for it's exclusive use. This type of multiplexing is currently widely used for the distribution of analog media such as radio and television. Computer systems do not normally use FDMA because it does not support the burst data model well. FDMA does have an advantage over CDMA and TDMA in that more than one station can be transmitting at a time.

TDMA is a networking standard in which only a single station on the network is allowed to transmit at a time. TDMA systems impose an order on the systems in the network and each station is explicitly given permission to transmit and explicitly releases the resource of the network. TDMA has many advantages for real time applications; most real time networking systems in use are TDMA based.

Star networks are also commonly used for time critical applications. These networks give a dedicated channel to each node in the network and employ a complex switch in the center of the network to manage the traffic. Examples of this type of application are ATM and switched Ethernet.

Here is a concise summary of the competing schemes that were considered for this project:

	Real time	Safety Critical	Low Cost
FDMA	Yes	Yes	No
TDMA	Yes	No	No
CDMA	No	Yes	Yes
Switched	Yes	No	No

From this we can conclude that none of the existing networking paradigms meet all of the constraints that we have proposed for our networking system. We could relax the constraints until one of the competing paradigms is clearly better.

Out thinking was as follows: the only scheme that has low enough hardware costs is CDMA in the form of Ethernet, so we are compelled to choose it as the hardware platform. Ethernet at it's lowest form is merely a method of putting information onto a wire and retrieving that information, so it should be possible to use this as a basic mechanism. It will then be possible to implement a TDMA protocol on top of that, resulting in a hybrid design using the low cost hardware of Ethernet, but abandoning the CDMA paradigm in favor of TDMA. This will result in a low cost solution that has the appropriate real time and safety critical characteristics. By using Ethernet hardware we also gain the ability to choose 10, 100, or 1000-megabit speeds and the ability to broadcast or multicast.

IMPLEMENTED HARDWARE

Custom hardware was needed to achieve the necessary real time requirements. The design had to be simple, low cost, and powerful enough that it would be capable of accomplishing the task that was being set out for it.

At the center of the design is a micro-controller a Motorola Coldfire. It has two serial ports and several timers as well as a small amount of memory. It also has much of the required hardware for interfacing to memories, and computer buses are built into the package We also use a Programmable Gate Array (FPGA) to add computational power to the system. As the main function of our system is to collect and play audio data on either of two speakers, a pair of audio codecs are also included.

The final, and possibly the most important element is the communications interface. Because we are implementing a TDMA scheme over an Ethernet, we chose a AMD-79C940BKC-MACE Ethernet Controller. It is a fully functional very inexpensive Ethernet interface chip, and it's possible to drive the Ethernet interface using interrupts.

In the implementation the network protocol the design does use 802.3 (Ethernet) compliant packets. However, packet sizes were fixed to carry 56 bytes of data, and no additional protocol information was included except for the minimum that was required. We do not need collision detection or re-transmission, but the design must avoid error conditions generated by the Ethernet hardware. A hardware error of concern is the jabber condition.

Ethernet packets have a maximum length; they can at most carry 1500 bits of payload data. Packets that exceed this maximum are considered to have "jabbered" on the network. An improperly implemented Ethernet driver, a hardware fault or more likely the packing too close of packets on the network medium can cause this.

DESIGN-IMPOSED CONSTRAINTS

In the calculations outlined in the previous sections we assumed values for two parameters: the inter-packet gap and the jitter. Now we have an implementation of this system and can measure these values directly.

Using the oscilloscope the minimum inter-packet gap was measured to be 0. This figure represents the closest that two packets may be safely place next to one another on the network before they are detected as a collision.

A zero value is optional according to the Ethernet standard, and the application controls timing well enough to permit it in practice.

The jitter in the system is the variance in time that the system shows over number of iterations, and was measured, as 0.023 ms. in the system as implemented this is the amount of variability that needs to be allowed for so that the systems can stay in synchronization. If this allowance is violated there is a possibility that two stations may transmit at the same time resulting in a fatal network collision.

The protocol is also extendable. Several enhancements could be made to the design that would significantly improve the utility and not violate the desirable real time properties. We could extend the system to support faster networking mediums such as 100-megabit and 1000-megabit Ethernet standards, which would allow a significant reduction in latency. The extension into a wireless or non-Ethernet optical system is also a reasonable extension.

HIGH LEVEL PROTOCOL DESIGN

The primary challenge for this network protocol is to ensure that no two stations transmit at the same time and each station gets a timely opportunity to transmit. There are two basic methods of implementing a TDMA protocol. First, a token may be passed around in the network and whichever station is in possession of the token may have exclusive access to the network; this is an effective system for sharing network resources and is the system in use for such systems as 802 token ring and FDDI networks. The problem with this type of system is that token passing introduces an unbounded latency into the system. The initial generation of and the regeneration of a lost token can take a long time and can be complex. Since it is a stated goal of the system to have bounded latency, this would not be an acceptable solution.

Second, it is possible to divide the network up into a series of time slots, each station being given a set of slots that it may transmit in. This system can be controlled by interrupt generating timers, commonly found in computer systems. This type of system does not have an overhead in processing a token, or the required complexity of token generation or regeneration. The essential difference between these two TDMA paradigms is that the token passing method does guarantee that each station will get the opportunity to use the network, so there is no guarantee concerning how long a node will have to wait before being given access. The windowing method does guarantee that every station that is on the network will get a specific amount of, bandwidth within the network latency. However there is a limit on the number of stations that can be present on the network. In this windowing method the individual systems in the network do need to have precisely synchronized clocks. The "broadcast and average" algorithms, such as NTP, do not provide sufficient accuracy to synchronize the transmission of packets in the microsecond range. Systems that can synchronize clocks to the required precision such as those based upon using a GPS receiver would require expensive hardware at each node and are not reliable enough to be used in safety critical applications. The solution to this dilemma is that since the clocks in the stations only need to be in sync relatively, the absolute timescale, as would be provided by a GPS based system, is not relevant for the application of these systems. A single node on the network that broadcasts a synchronization or "mark" pulse on the network that causes all stations to reset their internal timers to a relative time of zero is all that is required.

Given that the interval between mark pulses is the latency of the network it is now possible to construct a model of the network. There is a relationship between the latency, bandwidth and maximum number of time slots available, and therefore the number of stations.:

$$S = \left\lfloor B_T \Big/ \left(\frac{C \times D}{L}\right) \right\rfloor$$

Equation 1: Relationship between Slots and Network Parameters

where S is the number of slots available, L is the latency (frequency of mark pulse), BT is the bandwidth (bits per second), C is the sample per slot, and D is the bit length of the sample. The nodes are coordinated by a central or master node that sends out a synchronization or "mark" pulse. Each node in the network is given a specific fixed time period after the mark in which it may transmit.

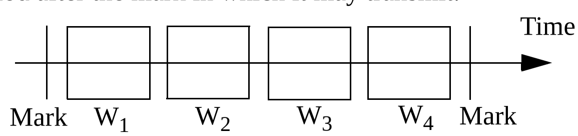


Figure 1: Example mark pulse as assigned windows.

The access to the network by each node is by each node being given exclusive access to the shared medium. Since the total amount of bandwidth is limited and each node in the network is pre-assigned a specific amount of it the total number of transmitters in the network is also limited. By changing the amount of delay between mark pulses and the amount of time that each node is permitted to transmit the, latently, bandwidth and number of stations that can participate in the network can be controlled.

The desired properties of the protocol are, bounded latency, guaranteed bandwidth and that is robust and reliable. The protocol described does meet all of those criteria.

Participating in the network there are two types of network station: a master station, that is responsible for periodically issuing a mark pulse, and the nodes participating in the communication. The control loops of each of these types of nodes can be described in pseudo-code.

Master Node - p code:

Task 1:

On the event of the mark timer going off.

Reset mark pulse timer.

Send mark pulse

Repeat

The client node can similarly be described:

Client Node - p code:

Task 1:

Wait for mark pulse to arrive

From the local node number calculate

time of transmission

window

Set transmission timer for window.

Task 2:

On the event of the transmission timer going off

Send data.

By assigning the responsibility of coordinating the participating nodes to a central controller we alter the access control model of standard broadcast mediums, such as Ethernet. This centralized control allows us to impose the necessary real time properties on the protocol.

A PROTOTYPE SYSTEM

A concrete example of a problem for which this protocol is suitable is the development of a simulation of an air traffic control audio system, with the expectation that the paradigms developed here could be used as template for an operational air traffic control system and other interactive multimedia systems. Air traffic controllers have very demanding requirements. They will need to listen and communicate with a variety of sources even is relatively simple circumstances. For example it would not be unusual for a controller to be listening to several air traffic radios and communicating with other air traffic centers over ground lines. Each of these audio sources needs to be presented to the controller clearly and in a timely fashion. Moreover there can be no delay and a minimum of interference over those audio channels.

A related application is the multiple participant virtual world environments where very low latency and guaranteed delivery is a requirement. Such an environment might include multi-participant virtual aircraft simulation, battlefield environments and other applications. For example this protocol is suitable for use in a virtual environment where all of the participants can hear and interact with each other in a highly reliable manner.

A non-obvious application that the storage industry is interested in a building a community of cooperating storage devices collected together over a dedicated local area network. The characteristics of the protocol make it suitable for building highly reliable, redundant and distributed storage systems.

The protocol that is to be implemented is to support ATC voice terminals. ATC voice terminals have a user interface of and transmit and receive audio data from other terminals and sources. The general requirements for the ATC system are: to provide telephone quality audio to each station; there can be no perceivable delay in communicating between any combination of systems on the network; and, the system will not introduce artifacts into the audio. For a minimal system there must be at least 8 audio sources that can be handled by each station. Those general requirements translate into specific requirement of supporting bi-directional 44.8Khz audio (56 samples over every 7 milliseconds) at each position. The system overall must have a maximum latency of less than 200 milliseconds. The system is to provide for the guaranteed delivery of data as well as guaranteeing that each node is serviced frequently enough so that the latency requirements are met. The system must also be deterministic, in that the calculated values must not change in the face of system load. This is a real world problem that this work is attempting to address.

As with many real world problems there is also the requirement of cost. The solution to this problem is to be kept to a minimum cost. Cost in these types of projects comes in many forms. First there is the cost of the time taken to implement and integrate the solution. Then there will be the cost of maintaining and improving the solution, and, in the case of commercial devices, there is the cost of building and deploying the solution. All of these cost factors will have to be taken into account in the design phase of the solution to the problem.

To summarize the networking requirements of a ATC audio network:

Bandwidth per station:	44.8Khz per station bi-directional constant data
Maximum system Latency	Sub 200 mS overall.
Reliability	Data must not be lost or corrupted in the transmission process.

Possible to detect the failure of either the position node or central master node.

The design of the system includes having a central node that provides the synchronization pulse to be broadcast on the network; this node is also to be responsible for combining or mixing the required output audio signals,. not allowing for cabling delays which are negligible. The relative time on each of the position nodes will then be precisely synchronized. Each station will be allocated a selection of time-slots that during which it is expected to transmit, and series of time slots that data for it will be available off the network. The initial synchronization pulse is called the "mark" pulse, which indicates the beginning of a new time window and the termination of the previous one. The window of time between two successive mark pulses is latency for a packet in the system, for example a mark pulse generated at 8Kz will result in latency of 7 milliseconds for a station to have the opportunity to both transmit and receive. Each transmission and receive window is guaranteed bandwidth to each of the stations. Thus both latency and bandwidth become a function of the frequency of the mark pulse.

So, using a 10 megabit internet and 128 bit packets:

$$S = \left\lfloor B_T \Big/ \left(\frac{C \times D}{L}\right) \right\rfloor$$

$$S = \left\lfloor (10000000) \Big/ \left(\frac{56 \times 8}{0.007}\right) \right\rfloor$$

$$S = 156$$

Equation 2: Calculation of Network Parameters

.A significant advantage this protocol has for real time systems in general and specifically for safety critical systems, is that the failure of individual nodes can be passively detected, even for the central node providing the mark pulse. Merely by collecting a profile of the network an extra, redundant station, can detect the failure of a station an take over for it without user intervention.

Several practical issues become apparent when looking at this design of the protocol. There are to be two distinct types of nodes in the network, a master node that provides the synchronization "mark" pulse to all other nodes in the network, and nodes that receive the pulse, and then select an appropriate window for them to transmit in. The arbitration of which node gets to transmit in what time slot is done before the starting of the network, each node in the system having a position in the network indicated by a small integer value. For example the station given the third position in the network would receive the third set of transmission windows.

To examine how this protocol will work in practice the best way is to image that we can see the frames being put onto the Ethernet wire over an interval between mark pluses. The theoretical model above does not allow for some real world concerns. Ethernet hardware normally detects the occurrence of a collision, and in-order to do this it there does need to be a brief period of network silence at the end of each packet. This value we will determine experimentally. The additional practical factor is jitter. Jitter is the delay in processing an interrupt on the system, and affects how closely the packets can be placed together on the network. The jitter in the network protocol is a summation of a number of factors:, the drift in the timers, if the interrupt handler are in cache memory, the length of cable between nodes, and the design of the code; these factors all add up to the jitter observed in the system. The jitter in the system does significantly affect the performance of the protocol.

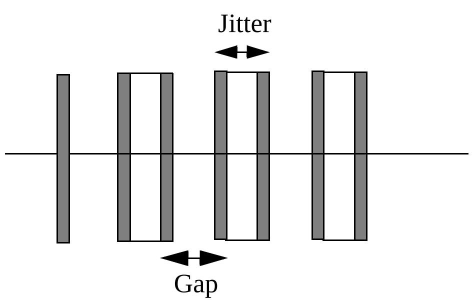


Figure 2: Jitter and gap defined.

The safety critical features of the system also need to be designed at this point. The primary type of failure that the safety critical aspects of the system are attempting to cover is the complete failure of a node. Because of the broadcast nature of Ethernet and the highly structured nature of the network protocol it is possible to passively detect when a station has gone off line. It is also possible to even detect the failure of the master node in the same manner. To ensure the safety critical features of a system implemented with this protocol it is necessary to design nodes that merely listen and wait for a node of a particular type to go off-line and to simply step in and take it's place.

The ATC system has the following model. Each of the position nodes collects audio data, sends it to a central node where it is processed and mixed together, the central node then sends out all of the resulting audio to the position nodes where the audio is played on a combination of speaker and headsets.

In the network model we have designed here there is no direct data transmission between position nodes; data is only transferred between the central, mixing node, and the position nodes. To that end we divide the time window in half, the first half is for the position nodes to transmit data to the mixer, the second half for the mixer to send the output data back to the positions to be played.

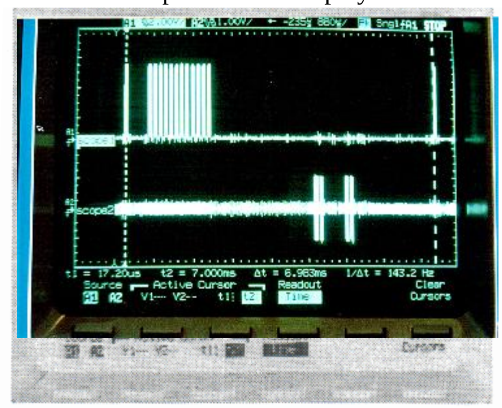


Figure 3: Network trace of Air Traffic Control (ATC) system.

The central master/mixing node co-ordinates the actions of the position nodes, controlling when they are permitted to transmit. The position nodes each receive an individual data packet for each interval from the central node.

The detailed design of the system is presented in the form of pseudo code. There are several different tasks outlined and they will all run on a variety of nodes in the system. There are detailed designs for the tasks on the master and position nodes, as well as designs for the backup master and position nodes. The design of the actions of the nodes in the system can best be understood in the tasks that they have to perform.

ATC Master Node – p code.

Task 1:

On the event of the mark timer going off.

Reset mark pulse timer.

Send mark pulse

Repeat

Task 2:

On the event of the mark pulse going off

Set transmission timer for 1/2 of the mark period

Repeat

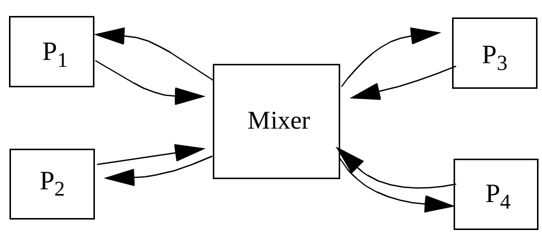


Figure 4: Central master/mixing node and data flow in ATC system.

Task 3:

On the event of the transmit timer going off

Send all frames to position nodes.

Task 4:

Do any necessary housekeeping functions.

ATC Position Node - p code.

Task 1:

Wait for mark pulse to arrive

From the local node number calculate time of transmission window

Set transmission timer for window.

Task 2:

On the event of the transmission timer going off

Send frame to audio mixer node

Task 3:

Perform any necessary housekeeping functions.**ATC**

It is also possible at this point to describe the automatic fail-over system the system can use. Essentially in the network there will be a number of nodes that listen passively for one of the active nodes to fail, and then take over the responsibilities for that node.

ATC Backup Master Node - p code

Task 1:

Wait for mark pulse to arrive

Re-set watchdog timer for slightly longer than mark period.

Task 2:

On the event of the watchdog timer going off

Restart node as master node.

From examining this design we can see that each node will be allowed to meet the real time constraints that have been laid out for it. The primary challenge for this type of system is to ensure that no two stations transmit at the same time and each station gets an opportunity to transmit; the design presented here does. It has also been seen that the broadcast nature of Ethernet can be an advantage for the purposes of synchronization and redundancy. This design requires that at least two different types of nodes be used in the network. The master node is responsible for providing the synchronization or "mark" pulse to the network. The position nodes use the reception of the "mark" pulse to synchronize the relative times between them and to select unique transmission windows. The tasks required for each type of node to implement the network protocol have been outlined. Each type of node has can be backed up by listening for the failure of a running node. The safety critical aspects of the system have been addressed by outlining the design for backup nodes as well.

CONCLUSIONS

We have successfully designed and implemented a protocol with good real time properties. The protocol developed here is tunable; it is possible to change the parameters of the protocol and trade one positive aspect for another. For example it is possible by increasing the latency in the protocol to increase the data bandwidth available to each node. We can also increase the latency, or decrease the bandwidth to increase the number of permitted nodes on the network.

A simple robust design can support the demands placed upon it. A fundamental characteristic of good design is that it should be no more complicated than necessary. A complex design is much more likely to fail simply by virtue of the fact that there will be more parts to fail. The protocol detailed here is an example of a design that is sufficiently complex to solve the task laid out for it and does not suffer from features that could interfere with its operation.

This work not only develops a theoretical model of a real time network system it provides an example implementation as well. We measured real performance compared to theoretical values for the particular network hardware that was used as well as confirmed the implementation of many aspects of the hardware.

We have created a simple, well designed, largely media independent network protocol that meets real-time requirements functions on cost-effective hardware.

REFERENCES

1. Teledisc Corporation, "Does Latency Matter", December 1997.
2. Andrew S. Tanenbaum, "Computer Networks". Prentice Hall Inc. New Jersey.
3. G. Kramer, B. Mukherjee, and G. Pesavento, “Interleaved Polling with Adaptive Cycle Time (IPACT): A Dynamic Bandwidth Distribution Scheme in an Optical Access Network,” Photonic Network Comm., vol. 4, no. 1,2002, pp. 89-107.
4. Su-il Choi and Jae-doo Huh, Dynamic Bandwidth Allocation Algorithm for Multimedia Services over Ethernet PONs, ETRI Journal, Volume 24, Number 6, Dec. 2002
5. Philip Irey, Robert Harrison and David Marlow, "Techniques for LAN Performance Analysis in a Real Time Environment", Journal of Real-Time Systems, vol 14, pp. 21-44.
6. Motorola - Consumer Systems Group, "Product Brief MFF5307 Integrated Coldfire Microprocessor," Motorola Document #MCF5307/D. 1997.

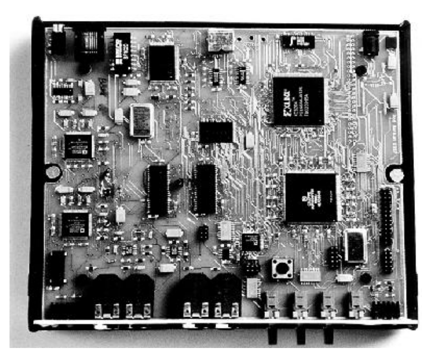


Figure 5: Real Time Network Platform

Adaptive Joint Source-Channel Coding for Real-Time Video Applications over Wireless IP Networks

Qi Qu[*], Yong Pei[‡], and James W. Modestino[†]

[*]Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, CA92093

[†]Department of Electrical and Computer Engineering, University of Miami, Coral Gables, FL33124

[‡] Department of Computer Science and Engineering, Wright State University, Dayton, OH45435

Abstract -- In this work we consider the delivery of H.264-coded video over future 3G wireless IP networks and we propose an adaptive motion-based unequal error protection (UEP) video coding/transmission system which can exploit knowledge of the source material as well as the channel operating conditions. Given this information, the proposed system can adaptively adjust the operating parameters of the video source encoder and the forward error correction (FEC) channel encoder to maximize the delivered video quality based upon both application-layer motion estimates and link-layer channel estimates. We demonstrate the efficacy of this approach using the ITU-T H.264 video source coder and the results indicate that a significant performance improvement, measured in terms of end-to-end PSNR, can be achieved compared to representative non-adaptive approaches. Furthermore, we demonstrate that this adaptive motion-based scheme can provide comparable performance to an R-D optimized adaptive comparison system, but with substantially reduced computational complexity.

I. INTRODUCTION

Transmitting real-time digital video over 3G wireless networks while providing appropriate end-to-end quality-of-service (QoS) performance is one of the major design challenges in enabling seamless multimedia networking. The need exists for both video coding and transport schemes which not only provide efficient compression performance but also provide relatively robust transport performance in the presence of link error effects resulting in bursty packet losses over wireless networks.

It has become clear, however, that an efficient and robust video transmission scheme requires a judicious combination of several error-resilience techniques which can be implemented from either a source or channel coding perspective. However, in almost all of the existing literature, error-resilience techniques have been employed separately and independently, although substantially improved performance can be achieved by an appropriate combination of such tools. In [3, 4], the approach of using "smart" inter/intra-mode switching based on an R-D analysis was presented. Unfortunately, such schemes result in a highly complicated system design/implementation, and their effectiveness in handling bursty packet losses is not clear. Furthermore, it is not apparent how they can be combined with FEC approaches. In [1, 2], the use of adaptive forward error correction (FEC) coding approaches based on the use of either source or channel information, respectively, was investigated and the efficacy of the corresponding adaptive FEC coding schemes was demonstrated. However, how to adaptively combine FEC coding with other mitigation schemes, such as passive error concealment (PEC) and intra-updating, and how effective the combination will be were not considered in these previous works.

As previously discussed, it is obvious that techniques from both the source and channel coding/decoding perspectives should be integrated and adaptively coordinated to optimize performance in response to time-varying source/channel dynamics. In this work, we propose an adaptive motion-based video coding/transmission system which can exploit the available source/channel dynamic information and provide an appropriate adaptive response to achieve near-optimal performance[1]. Facilitated by an application layer framing (ALF) perspective [6], the proposed adaptive scheme employs both adaptive intra-updating rate selection, used in combination with PEC, and the adaptive application of FEC coding. Our simulation results demonstrate that a substantial performance gain can be achieved by using this adaptive motion-based FEC/UEP approach compared to representative non-adaptive schemes. Furthermore, as we demonstrate, the results also indicate that the performance achieved by the proposed scheme is very close to that of a comparison model-based system which attempts to adaptively optimize the R-D performance.

The organization of this paper is as follows: in Section II, we provide some technical preliminaries. In Section III, we discuss the underlying motivation for the proposed adaptive motion-based FEC/UEP scheme based on previous work in [7]. Section IV presents the details of the adaptive motion-based system. In Section V, we provide simulation results and compare the performance of the proposed adaptive scheme with both a representative non-adaptive approach and an adaptive R-D optimized comparison system. Finally, a summary and conclusions are provided in Section VI.

II. PRELIMINARIES

A. RTP-H.264 and Network Model

In order to transmit real-time H.264 video over 3G wireless IP networks, the H.264 coded bit-stream has to be packetized as described in [9]. Based on the RTP-H.264 payload format specification, the encoded bit-stream is packetized and then transmitted as RTP/UDP/IP packets. Generally, the associated RTP/UDP/IP header is 40 bytes; however, in order to facilitate real-time applications over bandwidth-limited wireless networks, we use robust header compression (RoHC) to compress the RTP/UDP/IP header to 3 bytes [8].

[1] By optimal, we mean here the best performance in terms of reconstructed video quality, measured in terms of PSNR, achievable under the imposed design constraints.

The packet losses over wireless networks will most likely occur in bursts with possibly varying burst lengths. In this paper, we will use a Gilbert channel [11], to model the network-layer packet-loss characteristics of the wireless network. In the good state, the packet is transmitted and received correctly with probability 1, while in the bad state, the packet is discarded with probability 1. The Gilbert channel model can be uniquely specified by the packet loss rate (P_L) and the average burst length (L_B) measured in packets. They can be related to the state transition probabilities p and q as indicated in [11].

B. Error-Resilience Techniques

In this subsection, we briefly discuss three mitigation techniques employed in this paper: PEC, periodic intra-updating and FEC coding. A major focus of this paper is to determine and characterize a methodology for optimally combining such techniques while minimizing complexity.

For the PEC, if the packet is considered lost, the RTP sequence number enables the decoder to identify the lost packets, so that the locations of the missing slices which are composed of several MBs are known. All correctly received slices of a picture are decoded first, and then the lost slices are concealed according to the motion-compensated error concealment algorithm presented in [10]. Such PEC can be very effective in improving the objective and subjective quality of reconstructed video in the presence of packet losses, especially for low-motion sequences with little scene changes. However, the bursty nature of the packet losses may also degrade the effectiveness of this motion-compensated PEC strategy because successive packet losses are more likely to result in the loss of neighboring motion vectors which, in turn, decrease the accuracy of the error concealment.

As for the intra-updating, simply speaking, by using intra-coding for certain MBs in the video frame, the dependence of the reconstructed MBs on preceding frames can be terminated and error propagation effectively eliminated. However, the robustness provided by intra-updating comes at some expense, as it generally requires a higher bit rate than the more efficient inter-coding to achieve the same reconstructed video quality. So how to balance the error robustness achieved by intra-updating with the resulting reduction in source coding efficiency is an important issue. In this paper we study the effectiveness of a low-complexity periodic intra-updating scheme instead of a R-D based inter/intra-mode switching approach as described in [4]. More specifically, in the approach taken here, one slice in every several consecutive frames is intra-coded to enhance the error resilience capabilities in the face of packet losses. The specific intra-updating rates used are summarized[2] in Table I.

Finally, in general, a variety of FEC coding schemes can be applied with H.264 as discussed in [9]. In this paper, we use interlaced Reed-Solomon (RS) encoding as described in [2], [7]. Basically, for an RS(n, k) code, the erasure-correcting capability is $n - k$ given the locations of the lost packets are known. Each of the n packets is encapsulated as an RTP/UDP/IP packet to be transmitted over the wireless IP network. Considering the stringent delay constraints for real-time video services, it is desirable to keep the delay introduced by interlaced RS coding to within a single video frame. Since in this work each QCIF frame is assumed to be composed of 9 slices, this suggests use of RS(n, 9) codes. However, it should be noted that the use of FEC coding clearly introduces additional overhead which increases the actual transmission rate. On the other hand, use of larger values of n can provide improved erasure-correcting capability but at the expense of excessive overhead which reduces the bit-rate available for source coding. In what follows, we make exclusive use of the primitive RS(15, 9) code and punctured versions of it resulting in a class of RS(n, 9) codes with $9 \le n \le 15$ unless stated otherwise.

Table I. Intra-updating rates employed.

Without Intra-Updating	No Intra-Updated slice used
Medium Intra-Updating	1 Intra-Updated slice/2 Frames
High Intra-Updating	1 Intra-Updated slice/1 Frame

[2] While more finely quantized intra-updating rates are of course possible, the relatively coarsely quantized rates considered here are sufficient to demonstrate the efficacy of the proposed combined approach.

III. Motivation For An Adaptive System

Based on the results in [7], for wireless video applications, three distinct factors can greatly affect the system performance and would result in a different combination of error-resilience techniques to optimize performance in response to varying conditions. The first factor is the prevailing *channel loss condition*. If the channel conditions are poor, some degree of intra-updating and FEC coding is required in order to recover transmission errors and eliminate the error-propagation effects in video decoding. In this case, the benefits in terms of error robustness outweigh the performance degradation in source coding efficiency due to the use of intra-updating and FEC. However, if the channel conditions improve, the use of strong intra-coding and FEC coding may be detrimental to the overall system performance since now the source coding performance degradation becomes a dominant factor. The second factor is the *motion level of video frames*. The loss of a low-motion frame/slice is barely noticeable in the reconstructed video since it can be effectively concealed by the built-in error concealment. However, the loss of a high-motion frame/slice may cause substantial performance degradation due to the severe error propagation that results since error concealment cannot perform effectively in this case. Thus, stronger intra-updating and/or FEC coding become desirable, if not absolutely necessary, with increasing motion levels. Finally, the last factor would be the operating *bit-rate regime*. The use of intra-updating and FEC both have a detrimental effect on source coding performance, although the impact depends on the operating bit-rate regime. In particular, for low bit-rate regimes the loss in source coding performance becomes a dominant factor and aggressive intra-updating and FEC are not desirable, even when channel conditions are poor. For the high bit-rate regime, on the other hand, the degradation of source coding performance is less of an issue since bandwidth is plentiful. In this case, more aggressive intra-updating and FEC can be applied, although the appropriate combination will depend on channel conditions.

Based on these observations, it should be clear that in order to achieve optimum performance under the imposed design constraints, an adaptive system is required that, in addition to the use of PEC, chooses the proper intra-updating rate and the appropriate channel coding rate based upon information concerning the motion level in the source sequence together with information on the prevailing channel conditions, as well as the operating bit-rate regime.

IV. Proposed Adaptive Motion-based System

A. System Description

The proposed adaptive motion-based system illustrated in Fig. 1 is composed of an H.264 video encoder/decoder, a packetizer/depacketizer, an FEC encoder/decoder, a motion-level classifier and a channel condition estimator. The network characteristics are modeled in terms of a two-state Gilbert model as discussed previously.

Basically, we propose a motion-based scheme in which the video encoder adaptively selects the intra-updating rate for a sequence of N contiguous frames based on the estimated motion-level of these N frames, knowledge of the operating bit-rate regime and feedback information on channel conditions. After the video sequence is encoded by the H.264 encoder and packetized, FEC coding is then adaptively applied to each frame by likewise jointly considering the relative inter-frame motion between frames, the operating bit-rate regime and the prevailing channel conditions. In this work, a fixed packetization approach is employed which is the same as described in [7].

At the transmitter, the motion-level classifier is used to determine the inter-frame motion level of a sequence. The motion information is then provided to the video encoder, the FEC encoder and optionally to the packetizer. This information is used, as described below, by the video encoder to control the adaptive selection of intra-updating rate. Likewise, it is used by the FEC encoder to control adaptive FEC code selection.

At the receiver side, as illustrated in Fig. 1, we assume that a reliable low bit-rate feedback channel from the decoder to the encoder is available, which allows reporting an estimate of the channel characteristics (P_L&L_B), as observed at the decoder, with low latency. Under typical conditions, it can be assumed that the reverse channel is error-free and the associated overhead is negligible. Thus, the observed channel characteristics (P_L&L_B) at the decoder can be provided to the encoder using the RTCP protocol which can enable reasonably continuous feedback concerning the overall reception quality at the receiver.

B. Implementation Issues

1.) Motion-Level Classifier

The key component of this adaptive motion-based system is the motion-level classifier, which is used to determine the motion-level information of an input video sequence. In previous work [12], motion has been estimated by a statistical classifier; namely, by the inter-frame prediction error which is used as a primary indicator of activity of video frames. The results in both [2] and [12] have demonstrated that this simple way to determine the inter-frame motion or activity of video

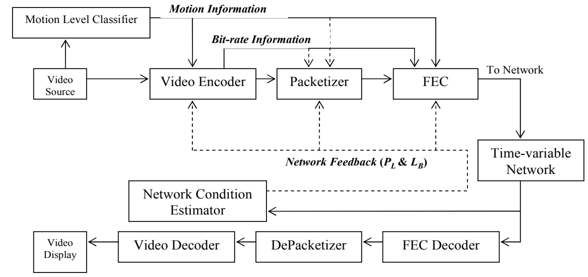


Fig. 1. Motion-based adaptive system configuration.

frames using the inter-frame prediction error is effective. Therefore, we will use this statistical classifier in our adaptive motion-based system. In this paper, we do not differentiate between I-frames, P-frames or B-frames; the same operations are performed independently of the frame type.

In the motion-level classifier, we first calculate the mean-square inter-frame prediction error between adjacent frames according to

$$E[m] = \sum_{j=0}^{N_v-1} \sum_{i=0}^{N_h-1} \left[X_m(i,j) - X_{m-1}(i,j)\right]^2, \tag{1}$$

where $E[m]$ denotes the mean-square inter-frame prediction error between the m-th frame, of size N_v x N_h pixels, and the $(m-1)$-th frame of the video with the same size. Again, the quantity $X_m(i,\ j)$ represents the luminance values at pixel position (i, j) in frame m.

2.) Adaptive Operations in the Video Encoder

Since several adjacent frames in a video sequence usually have correlation to some extent, it is reasonable to estimate the motion level of N contiguous frames by using the average of the mean-square prediction errors of the previous N frames with an appropriately selected N. In order to acquire an estimate of the motion level for the current N adjacent frames, which is necessary for selection of the proper intra-updating rate at the video encoder, a block-based estimate with a length of N frames is used to obtain the local estimate of the motion level of the current N adjacent frames (kN, $kN+1$, … ,$(k+1)N-1$) based on the average mean-square prediction error calculated for the previous N successive frames ($(k-1)N$, $(k-1)N + 1$,…,kN - 1) according to

$$Average = \frac{1}{N} \sum_{m=(k-1)N}^{kN-1} E\left[m\right], \tag{2}$$

where k is a positive integer used to indicate the order of each block with N frames.

By averaging the prediction errors of the previous N successive frames, we obtain a more reliable local estimate of the motion level over the current N contiguous frames. This estimate is used in the video encoder to control the adaptive use of intra-updating under different operating conditions.

In the motion-level classifier, we prestore a single motion-level threshold T_I which is used to classify the motion levels. In particular, by comparing the estimate obtained from (2) to this threshold, the N frames are classified into one of two motion levels: low-motion or high-motion. Then, the appropriate intra-updating rate is applied to the next N frames based on this classification. The block length N should not be taken too large to minimize the latency in the motion classification. At the same time, it needs to be consistent with the specific intra-updating rates used as indicated in Table I. Therefore, in the work described here we have taken $N = 2$. The preset threshold cannot be determined analytically. Instead, a series of subjective and objective tests were conducted on a variety of broadcast and teleconferencing sequences to determine appropriate empirically-determined values as described in [2]. This classification is used together with knowledge of the operating bit-rate regime and the channel conditions to determine which intra-updating rate should be used for the current joint conditions.

3.) *Adaptive Operations in the FEC Encoder*

Likewise, a threshold comparison of the estimated motion level for individual frames is used to determine whether or not that frame should be FEC encoded. More specifically, when the prediction error of a frame, as determined by (1), exceeds a preset threshold an appropriate choice of a RS code is applied to that frame; otherwise, no FEC coding is applied to that frame. However, rather than a fixed static choice of a threshold we employ a number of preset thresholds $T_F^{(1)} < T_F^{(2)} < T_F^{(3)} < T_F^{(4)} < T_F^{(5)}$, which we refer to as FEC thresholds. Similar to the motion-level threshold T_I, these FEC thresholds are also obtained from a series of subjective and objective tests.

The particular threshold in effect at any time is dynamically chosen on the basis of a joint combination of the current motion classification associated with that frame, the operating bit-rate regime and the prevailing channel conditions (i.e., P_L and L_B). The manner in which this dynamic choice is made will be made precise in what follows. For the time being, observe that corresponding to each choice of threshold $T_F^{(i)}$, i=1, 2,..,5, and selection of an RS code, the corresponding average channel coding rate (bits/c.u) can be calculated. For example, when an RS(n, 9) code is used, we find that $R_C \approx 1$ bits/c.u. if the threshold is chosen so large that very few frames are FEC encoded. Conversely, if the threshold chosen is so small that nearly every frame is FEC encoded then the corresponding average channel coding rate is $R_C \approx 9/n$. So, based on the preceding discussion, the selection and use of different FEC thresholds in the proposed adaptive motion-based scheme corresponds to the use of different effective channel coding rates. In particular, by selectively applying FEC only to those frames whose locally estimated motion levels exceed the corresponding threshold we provide UEP based on the importance of that frame to overall reconstructed image quality.

4.) *Adaptation Logic*

The adaptation logic implemented at the encoder is based on the use of two prestored tables, one each for low-motion and high-motion sequences. These tables have been obtained empirically and are used to instruct the video encoder and FEC encoder what operations to be performed based on the current combined operating conditions.

Once the first frame of a group of N successive frames enters the video encoder, the encoder compares the average mean-square prediction error calculated from (2) for the past N successive frames to the preset intra-updating threshold in order to determine the motion level of the current N successive frames. Thus, the appropriate operating table used for the current N successive frames can be determined. After entering the appropriate table, based on the associated information, the encoder first determines which intra-updating rate and which FEC threshold $T_F^{(i)}$ should be applied to that sequence of N contiguous frames. The local inter-frame prediction error obtained from (1) is then compared to the selected FEC threshold whose value is indicated in the next section. If the prediction error of a frame is larger than the selected threshold, appropriate FEC coding is applied; otherwise, no FEC coding is used for that frame[3]. The table sizes are relatively small and, utilizing fast searching techniques, this process will require only a small amount of time and memory.

C. Performance Evaluation

1.) *Complexity Analysis*

An advantage of the proposed adaptive motion-based FEC/UEP scheme is its substantially reduced computational complexity compared to alternative approaches. For example, in the R-D based inter/intra-mode switching scheme (ROPE) described in [4] it is required to compute the distortion and two moments for the luminance value of each pixel for the cases of intra-coding and inter-coding. As shown in [4], for every pixel in an inter-coded MB, it requires 16 addition/16 multiplication operations for the required calculation and for each pixel in an intra-coded MB, 11 addition/11 multiplication operations are required. By comparison, neglecting the additional complexity associated with the adaptive FEC coding and decoding, implementation of the proposed adaptive intra-updating approach in the video encoder requires only (N + 1) addition/2 multiplication operations for each pixel, regardless of the MB coding mode. In particular, if N=2 as used in this paper, then we need only 3 addition/2 multiplication operations for each pixel. Furthermore, the extension of the ROPE algorithm [4] to H.264 is not feasible due to the more advanced techniques employed in H.264 which require taking into account the expectation of products of samples at different positions to obtain an accurate estimation of the resulting distortion [9]. Thus, the computational burden is alleviated substantially in the proposed approach, even including the relatively modest computational cost for table look-up.

2.) *A Comparison System*

In order to provide a meaningful comparison between our approach and other alternative formula-driven optimization approaches, we considered use of a recent R-D optimized approach [5] as a comparison system. The R-D model associated with this approach can be used to estimate the decoder distortion and thus provide the optimization selection at the H.264 encoder. Again, the reconstructed video quality is affected by both source compression and quality degradation due to packet losses. In [5], it is assumed that the two kinds of induced distortion are independent and additive. Thus, we can calculate the overall distortion in terms of *MSE* as

$$D_d = D_s + D_c, \tag{3}$$

where D_d denotes the overall distortion; D_s and D_c denote the distortion induced by source compression and channel errors, respectively. And D_s and D_c can be computed as detailed in [5].

[3] Note that the same FEC threshold selection applies for the entire block of N contiguous frames and is updated on a block-by-block basis.

In the R-D optimized comparison system we use equations (3) to calculate the distortion for each pair of intra-updating rate and selected RS(n, 9) code, n = 9,…,18. Then, the pair which can achieve the minimum distortion is employed for the incoming N consecutive frames.

V. Simulation Results and Discussion

In this section, we present simulation results obtained by using the proposed adaptive motion-based FEC/UEP approach. First of all, we describe the simulation environment used in this paper.

Video sequences were encoded using the ITU-JVT JM3.9 codec of the newly developed H.264 video coding standard. In this paper, we first used two typical QCIF video test sequences, Foreman and Suzie, as described previously. Both are coded at constant bit rates specified by using the associated rate-control scheme [13]. The first frame of the sequence is intra-coded and the rest of the frames are inter-coded as P frames with/without slice-based intra-updating. We also employ a QCIF video test sequence which is obtained as a concatenation of the QCIF Carphone and QCIF Silent sequences to illustrate performance on a test sequence outside the training set. In our packetization scheme, each slice is packetized into one RTP packet; thus every QCIF frame is packetized into 9 RTP packets. For the adaptive motion-based FEC/UEP scheme, a fixed RS(15, 9) code is used selectively based on local motion estimates and the prevailing channel conditions which results in UEP for the transmitted video sequence. All the simulation results in this section have been obtained using 100 runs in order to obtain statistically meaningful average values.

In Fig. 2, we illustrate a plot of PSNR versus the packet loss rate P_L with $L_B = 4$ for the Foreman sequence at $R_{tot} = 96$ Kbps, which is in the range of low bit-rate services. The proposed adaptive scheme can be seen to achieve a much higher performance in terms of end-to-end PSNR compared to the representative non-adaptive schemes which apply the RS(15, 9) code to every frame, together with fixed intra-updating rates, regardless of its motion level and other source/channel conditions. Obviously, for lossless transmission (P_L = 0), optimum performance is achieved with no intra-updating and no FEC coding. The adaptive scheme achieves this by jointly considering the source/channel information. However, for the non-adaptive scheme, there is a considerable performance disadvantage due to the improper use of intra-updating and FEC/UEP coding. For example, for non-adaptive use of the RS(15, 9) code, together with a fixed high intra-updating rate, the performance loss is almost 7 dB compared to the adaptive motion-based FEC/UEP scheme. As P_L increases, the performance gap is reduced since when the packet loss rate is high, obviously more frames need to be FEC protected against packet losses. Therefore, for example, when P_L = 15%, the adaptive scheme has almost the same performance as the non-adaptive scheme with high intra-updating.

It should also be noted from Fig. 2 that the performance achieved by the adaptive motion-based scheme is very close to the performance of the R-D optimized comparison system, which is obtained by using the minimizing distortion pair among all the combinations of intra-updating rates and all RS(n, 9) codes, where n = 9,…,18. In particular, as can be seen from Fig. 2, the performance gap between our motion-adaptive scheme and the R-D optimized comparison scheme is less than 0.15 dB, and then only for large packet-loss rate. Although the R-D optimized comparison system can achieve a marginal performance gain compared to the propsoed scheme, the induced system complexity is much higher since: firstly, it requires a complex regression algorithm to calculate and update the model parameters continuously; secondly, for a given channel condition, the comparison system has to calculate the residual packet-loss rate for every RS code and it also needs to calculate the corresponding distortion for every combination of intra-updating rate and RS code. Therefore, considering the marginal performance gain of less than 0.15 dB, we can say that in real-time video applications over power-limited 3G system, the adaptive motion-based FEC/UEP approach is superior to the R-D optimized comparison system in terms of complexity while providing almost the same performance under the imposed design constraints. We will further demonstrate this point in what follows.

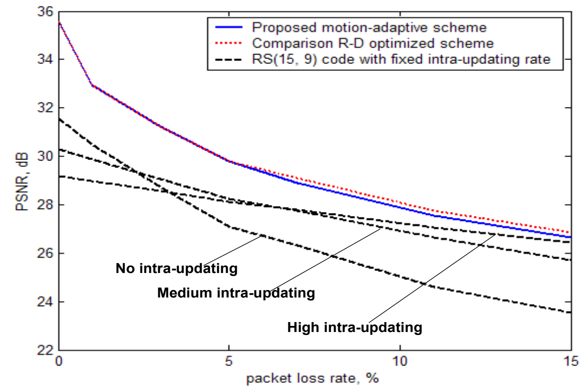


Fig. 2. Performance achieved by the adaptive motion-based scheme with an RS(15, 9) code, for the Foreman sequence at $R_{tot} = 96$ Kbps and $L_B = 4$.

As discussed previously, the Foreman sequence is a high-motion sequence, while the Suzie sequence is a low-motion sequence. In order to provide a more comprehensive evaluation of the proposed adaptive motion-based scheme, we repeat the results for the Suzie sequence in Fig. 3, for R_{tot} = 96 Kbps.

In Fig. 3, for the Suzie sequence at R_{tot} = 96 Kbps, observe that the relative performance gain compared to the non-adaptive approaches is substantial even at very high packet-loss rate, i.e., P_L = 15%. It can be seen that the performance gain using the adaptive scheme is more than 4 dB compared to the non-adaptive schemes at P_L = 15%, and when P_L is low, the gain is much larger. For lossless transmission, the gain achieved by the adaptive scheme is 7.5 dB compared to the non-adaptive schemes. The reason is that the Suzie sequence is a low-motion sequence with little scene and background changes, so that when the available total bit rate is low, here only 96 Kbps, optimum performance is achieved using no FEC coding, but using only intra-updating and PEC. The adaptive scheme achieves this by operating in a non-FEC coding mode. However, the improper use of FEC coding by the non-adaptive systems results in a substantial performance loss. Finally, the performance achieved by this adaptive scheme can be seen to be virtually identical to the R-D optimized comparison system.

In order to further demonstrate the effectiveness of this proposed motion-adaptive approach, we employed another video sequence at 30 fps rather than the QCIF Foreman and Susie sequences for which the adaptation logic was designed. It is constructed as a concatenated sequence of two other sequences as follows: the first 30 frames of the new video sequence is obtained from the QCIF Silent sequence at 30 fps, which has relatively low motion, and the remaining 30 frames is obtained from every third frame of the relatively high-motion QCIF Carphone sequence at 10fps. Thus, the first part of the concatenated sequence has a relatively low motion level while the latter part has a relatively high motion level. This concatenated sequence is then quite different from either of the two design test sequences.

By using the two fixed operating tables described previously, the corresponding adaptive operations are performed as indicated. Here, we assume the channel conditions are $L_B = 4$ with various P_L and $R_{tot} = 192$ Kbps. We compare the resulting performance with the performance achieved by the corresponding non-adaptive schemes and the R-D optimized comparison system in Table II.

From Table II, it is evident that the proposed adaptive approach can outperform the corresponding fixed schemes, especially when the packet-loss rate is low; while when the packet-loss rate increases, the adaptive approach performs only marginally better than the fixed schemes. This follows from the fact that as the channel conditions become more severe, the adaptive scheme tends to protect more frames with FEC coding and with higher intra-updating rate as explained previously. On the other hand, the proposed adaptive approach can achieve almost the same performance as the R-D optimized comparison system. The largest performance gap is about 0.3 dB which is still marginal. This table further demonstrates that our approach is effective not only for the training sequences which are used to design the system but also effective for the video sequences outside the training set.

VI. SUMMARY AND CONCLUSIONS

In this paper we proposed an adaptive motion-based FEC/UEP coding and transmission system for real-time video transmission over power-limited wireless IP networks. This proposed scheme can exploit the available source/channel information efficiently, i.e., the motion level of the transmitted video sequence, the operating bit-rate regime and the channel feedback which is assumed perfect. Thus, the scheme can adaptively respond to the time-varying dynamics of the source/channel behavior by adjusting the selected intra-updating rate at the video encoder and can adaptively apply FEC coding to frames where FEC coding is deemed necessary. The excellent performance that can be obtained by using this scheme is verified by simulation results which show that the performance gain is substantial compared to representative non-adaptive schemes. Also, as indicated by the simulation results, the performance achieved by this proposed adaptive scheme is very close to the performance of an R-D optimized comparison system subject to the same design constraints while greatly reducing the implementation and computational complexity, which is a dominant factor for real-time video applications in a power-limited wireless IP network.

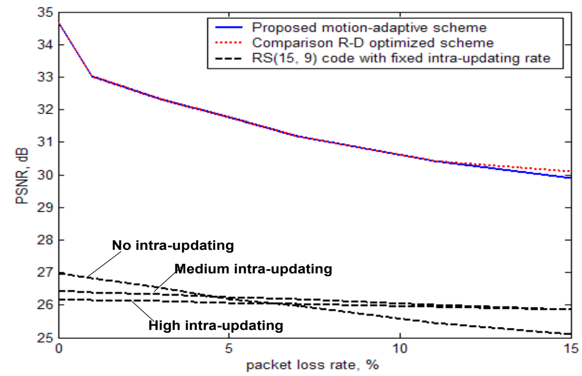


Fig. 3. Performance achieved by the adaptive motion-based scheme with an RS(15, 9) code, for the Suzie sequence at $R_{tot} = 96$ Kbps and $L_B = 4$.

TABLE II

Performance comparison (in dB) between the proposed adaptive approach, the non-adaptive schemes, and the comparison R-D optimized system for the concatenated video test sequence; $L_B = 4$ and $R_{tot} = 192$ Kbps.

	$P_L = 1\%$	$P_L = 7\%$	$P_L = 15\%$
Proposed adaptive approach	32.3	27.8	27.5
R-D optimized comparison system	32.3	28.0	27.8
RS(15,9)and medium intra-updating	29.0	27.4	27.0
RS(15,9)and high intra-updating	27.5	27.3	27.1

REFERENCES

[1] W. Kumwilaisak, J. Kim and C.-C. J. Kuo, “Video Transmission over Wireless Fading Channels with Adaptive FEC”, in *Proc. Picture Coding Symposium*’01, pp. 219-222, Seoul, Korea, Apr. 2001.

[2] M. Bystrom, V. Parthasarthy, and J. W. Modestino, “Hybrid Error Concealment Schemes for Broadcast Video Transmission over ATM Networks”, *IEEE Trans. on Circuits and System for Video Technology,* vol. 9, no. 6, pp. 868-881, Sept. 1999.

[3] Y-K. Wang, M. M. Hannuksela and M. Gabbouj, “Error-robust inter/intra mode selection using isolated regions”, in *Proc. of Packet Video Workshop 2003*, Apr. 2003.

[4] R. Zhang, S. L. Regunathan and K. Rose “Video Coding with Optimal Inter/Intra-Mode Switching for Packet Loss Resilience”, *IEEE JSAC*, vol.18, No.6, pp. 966-976, June 2000.

[5] K. Stuhlmuller, N. Farber, M. Link and B. Girod, “Analysis of video Transmission over lossy channels,” *IEEE Journal on Selected Areas in Communication*s, vol. 18, pp. 1012–1032, June 2000.

[6] D. D. Clark and D. L. Tennenhouse, “Architecture Considerations for a New Generation of Protocols”, *Comput. Commun*. Rev., vol.20, pp. 200-208, Sep.1990.

[7] Q. Qu, Y. Pei and J. W. Modestino, “Robust H.264 Video Coding and Transmission over Bursty Packet Loss Wireless Networks”, in *Proc. of IEEE VTC-2003/Fall,* pp. 3395 -3399, Orlando, Oct. 2003.

[8] S. Casner, V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links", RFC 2508, February 1999.

[9] T. Stockhammer, M. M. Hannuksela and T. Wiegand, “H.264/AVC in Wireless Environments”, *IEEE Trans. on CSVT*, pp. 657-673, July. 2003.

[10] Y.-K. Wang, M. M. Hannuksela, V. Varsa , A. Hourunranta, and M. Gabbouj, “The Error Concealment Feature in the H.26L Test Model”, *ICIP2002*, pp. 729-732, Sep. 2002.

[11] E. N. Gilbert, “Capacity of a burst-noise channel”, *Bell Syst. Tech. J.*, vol. 39, pp. 1253-1256, Sept. 1960.

[12] A. Puri and R. Aravind, “Motion-Compensated Video Coding with Adaptive Perceptual Quantization”, *IEEE Trans. on CSVT,* vol.1, pp. 351-361, Dec. 1991.

[13] S. Ma, W. Gao and Y. Lu, “Rate Control on JVT Standard”, Joint Video Team, 4th Meeting: Klagenfurt, Austria, 22-26 July, 2002.

An Asynchronous Multi-source Streaming Protocol for Scalable and Reliable Multimedia Communication

Satoshi Itaya†, Naohiro Hayashibara†, Tomoya Enokido‡, and Makoto Takizawa†
†Tokyo Denki University
Email: {itaya, haya, taki}@takilab.k.dendai.ac.jp
‡Rissho University
Email: eno@ris.ac.jp

Abstract

In a peer-to-peer (P2P) overlay network, a large number of peer processes are cooperating. Multimedia contents are distributed to peers in various ways like downloading and caching. In multimedia streaming applications, multimedia data is required to be delivered to processes in a real-time manner. In this paper, we newly discuss an asynchronous multi-source streaming (AMSS) approach where multiple contents peers transmit packets of a multimedia content to each requesting leaf peer. Here, each of the contents peers transmits different packets with some redundancy to the leaf peer in order to be tolerant of faulty contents peers or packets lost in networks.

1. Introduction

In peer-to-peer (P2P) overlay networks [10,16], a large number of peer processes (peers) are cooperating by exchanging messages in networks [1, 3, 5]. Multimedia streaming service [11, 13] is required to be provided for various types of applications like distance learning. Contents are in nature distributed in various ways like downloading. Not only high-performance contents servers but also peers can distribute contents in P2P networks.

One-to-one/one-to-many types of communication protocols like TCP [9] and RTP [12] are so far developed and widely used for multimedia applications. One-to-one and one-to-many protocols to satisfy Quality of Service (QoS) requirements like delay time, bandwidth, and packet loss ratio are discussed in papers [4,15]. Traditional streaming service is realized by using various types of communication protocol [8,9,12,13].

A peer which can provide other peers with a content is referred to as *contents peer*. On request of a content C, a contents peer transmits packets of the content C to the requesting leaf peer. Multimedia streaming applications are getting more significant in P2P environment. Large number of leaf peers have to be supported and every packet of a content is required to be reliably delivered to each peer so as to satisfy the real-time constraint. We discuss high-performance and highly reliable data transmission mechanisms named *asynchronous multi-source streaming* (AMSS) approach on P2P overlay networks. Here, multiple *contents* peers deliver multimedia contents to a leaf peer. Each of the contents peers transmits packets of the content to the leaf peer. In addition, data of the content are replicated by sending parity packets that the leaf peer can receive the whole content even if some number of contents peers are faulty and some packets are lost due to buffer overruns and network congestions. In our protocol, every active contents peer asynchronously starts transmitting a subsequence of the packets to each leaf peer independently of the other contents peers. Each contents peer autonomously selects packets to be transmitted by exchanging control information with other active contents peers in a group protocol [6,7,14], so that the active contents peers do not send same packets to each leaf peer and packets are transmitted in parallel.

In section 2, we present a system model of multi-source streaming on P2P overlay networks. In section 3, we discuss the asynchronous multi-source streaming (AMSS) protocol. In section 4, we evaluate the AMSS protocol in terms of throughput, packet loss ratio, and jitter delay.

2. Multimedia Streaming Models

2.1. Single-source streaming model

We take multimedia streaming applications [11, 13] like video on demand in a peer-to-peer (P2P) overlay network. P2P applications are realized by cooperation of multiple peer processes (*peers*) by exchanging multimedia data with other peers. Here, a *leaf* peer sends a request of a content C to a *contents* peer which supports the multimedia content C. Let $\mathbf{CP}_C$ be a set of contents peers $CP_1, \ldots, CP_n$ ($n \geq 1$) supporting a content C. Let $\mathbf{LP}_C$ be a set of leaf peers $LP_1, \ldots, LP_m$ ($m \geq 1$) requesting a content C. Then, the contents peer starts transmitting the content C to the leaf peer. In the traditional single-source streaming (SS) model, one contents peer supports multiple leaf peers and transmits packets of the content to each leaf peer asynchronously with the other leaf peers.

In order to realize the higher reliability and throughput of multimedia streaming service, multiple contents peers are used to deliver a content C to each leaf peer. Each leaf peer LP_s issues a request of the content C independently of the other leaf peers (s = 1, …, m). On receipt of a request of a content C from a leaf peer LP_s, a contents peer CP_i transmits a sequence of packets to LP_s. In one way, each leaf peer receives packets from one contents peer and another leaf peer may receive packets from another contents peer as shown in Figure 1. The load of a contents peer is distributed to multiple contents peers. However, if one contents peer is faulty, leaf peers serviced by the faulty contents peer cannot take continuous streaming service. In addition, if some packets are lost in networks, a leaf peer cannot take enough QoS of the content.

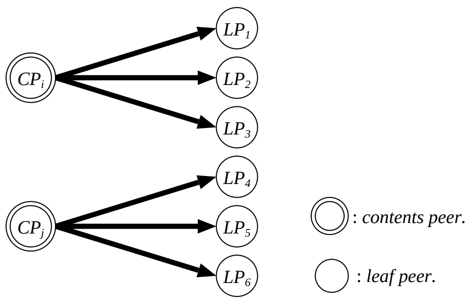


Figure 1. SS approach.

2.2. Multi-source streaming model

We propose a novel data streaming model named *asynchronous multi-source streaming* (AMSS) model to reliably and scalably deliver multimedia contents to a large number of leaf peers so as to satisfy QoS requirements, especially real-time requirement. Here, each leaf peer receives packets of a multimedia content C from multiple contents peers while each contents peer sends packets to multiple leaf peers as shown in Figure 2. Here, multiple content peers $CP_1, \ldots, CP_n$ send packets of the content C to a leaf peer LP_s as follows:

1. Parallel transmission of packets from multiple source contents peers to a leaf peer LP_s.
2. Minimally redundant transmission to LP_s.

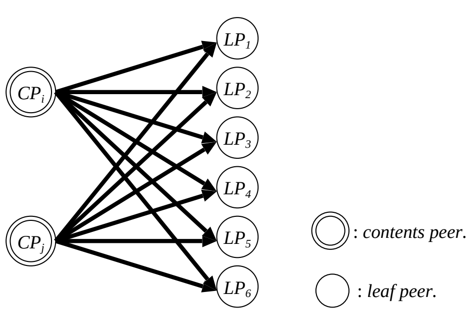


Figure 2. MSS approach.

In our approach, each contents peer CP_i does not send a same sequence of packets while multiple contents peers send packets in parallel. Each contents peer CP_i typically sends packets different from every other contents peer CP_k ($k \neq i$) to a leaf peer LP_s.

A contents peer CP_i decomposes a multimedia content C into a sequence pkt = $\langle t_1, \ldots, t_l \rangle$ of packets. Suppose a sequence pkt = $\langle t_1, \ldots, t_8 \rangle$ of packets is obtained from a multimedia content C. A contents peer CP_i sends a subsequence pkt_{is} of pkt to a leaf peer LP_s. For example, a subsequence pkt_{is} is composed of packets { t_h | h = i + $n{\cdot}d$ for d = 0,1, …}. That is, pkt_{1s} = $\langle t_1, t_4, t_7 \rangle$, pkt_{2s} = $\langle t_2, t_5, t_8 \rangle$, and ptk_{3s} = $\langle t_3, t_6 \rangle$. A *union* $pkt_1 \cup pkt_2$ of packet sequences pkt_1 and pkt_2 is a packet sequence including every packet in pkt_1 and pkt_2 where the packets are totally ordered in the sequence number and no redundant packets are included. Let $pkt\langle t_i]$ and $pkt[t_i\rangle$ show a prefix $\langle t_1, \ldots, t_i \rangle$ and postfix $\langle t_i, \ldots, t_l \rangle$ of a packet sequence pkt, respectively.

On receipt of a request of a content C from a leaf peer LP_s, contents peers CP_1, CP_2, and CP_3 send subsequences pkt_{11} = $\langle t_1, t_4, t_7 \rangle$, pkt_{21} = $\langle t_2, t_5, t_8 \rangle$, and pkt_{31} = $\langle t_3, t_6 \rangle$ to LP_s, respectively. Then, LP_s obtains the packet sequence pkt = $pkt_{11} \cup pkt_{21} \cup pkt_{31}$ = $\langle t_1, \ldots, t_8 \rangle$ from the subsequences pkt_{11}, pkt_{21}, and pkt_{31} sent by CP_1, CP_2, and CP_3, respectively.

A sequence pkt = $\langle t_1, \ldots, t_l \rangle$ of packets of the content C is stored in a local queue LQ_i of each contents peer CP_i (i = 1, …, n). Here, n is the number of contents peers. t := **dequeue**(Q) and **enqueue**(Q, t) show procedures where a packet t is dequeued from a queue Q and is enqueued into Q, respectively. Packets are distributed into n transmission queues $XQ_{i1}, \ldots, XQ_{in}$ as follows:

k := 0;
while($LQ_i \neq \phi$){
    k := k +1; t := **dequeue**(LQ_i);
    h := **Did**(k, n); **enqueue**(XQ_{ih}, t); }

A function **Did**(k, n) is realized to be $mod(k-1, n)$ + 1. Suppose there are three contents peers CP_1, CP_2, and CP_3. The first packet t_1 is enqueued into the transmission queue XQ_{i1} in the contents peer CP_i since **Did**(1, 3) = mod(0, 3) + 1 = 1. Packets t_2 and t_3 are enqueued into XQ_{i2} and XQ_{i3}, respectively, as shown in Figure 3.

send(t, P) shows a transmission procedure where a packet t is sent to a peer P. On receipt of a content request from a leaf peer LP_s, a contents peer CP_i sends packets as follows:

k := **Qid**(i, s, n);
while($XQ_{ik} \neq \phi$){
    t := **dequeue**(XQ_{ik}); **send**(t, LP_s); }

A function **Qid**(i, s, n) is realized as $mod(i+s, n)$ + 1. A contents peer CP_i transmits packets to the leaf peer LP_s from a transmission queue XQ_{ik} where k = **Qid**(i, s, n). The contents peer CP_i transmits another subsequence pkt_{ih} in XQ_{ih} to another leaf peer LP_t where h = **Qid**(i,

t, n). For three contents peers CP_1, CP_2, and CP_3, and a pair of leaf peers LP_1 and LP_2, **Qid**(1, 1, 3) = 3, **Qid**(2, 1, 3) = 1, and **Qid**(3, 1, 3) = 2. CP_1, CP_2, and CP_3 send packets in XQ_{13}, XQ_{21}, and XQ_{32} to LP_1, respectively.

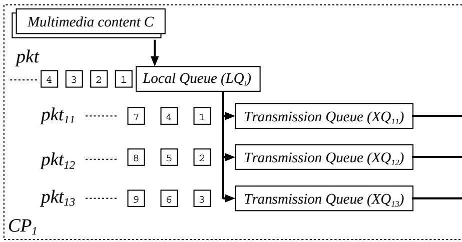


Figure 3. Distribution of packets.

3. Protocol

3.1. Data structure

Each data packet t is assigned with a unique sequence number $t.SQ$. A contents peer CP_i considers CP_j to be *active* if CP_i receives a control packet from CP_j. If CP_i does not receive any control packet from CP_j for some time units, CP_i considers CP_j to be *dormant*. Here, $\mathbf{CCP}_i$ is a view of CP_i, i.e. a subset of contents peers which CP_i perceives to be active. Initally, $\mathbf{CCP}_i = \{CP_i\}$ for CP_i. In each CP_i, the following variables are manipulated:

- SQ_i = sequence number of a data packet which CP_i has most recently transmitted, initially zero.
- SQ_j = sequence number of a data packet where CP_i knows that another CP_j has sent every data packet t where $t.SQ \leq SQ_j$ to the leaf peer LP_s, initially 0 ($j = 1, \ldots, n$, $j \neq i$).
- MVQ_{jk} = sequence number of a data packet where CP_i knows that CP_j has known that CP_k sent every data packet t where $t.SQ \leq MVQ_{ik}$, initially 0.
- $MVQ = \{MVQ_{jk} \mid j, k = 1, \ldots, n\}$.
- VW_i = view $\langle V_1, \ldots, V_n \rangle$ of CP_i where $VW_i.V_l$ = 1 if another $CP_l \in \mathbf{CCP}_i$, i.e. CP_i perceives another CP_l to be active, $VW_i.V_l$ = 0 otherwise.
- VW_j = view $\langle V_1, \ldots, V_n \rangle$ of CP_j which CP_i knows ($j = 1, \ldots, n$, $j \neq i$).
- $MinMVQ_j$ = sequence number where CP_i knows that CP_j has known that every active contents peer sent every data packet t where $t.SQ \leq MinMVQ_j$ ($j = 1, \ldots, n$, $j \neq i$).
- $MinMVQ_i$ = min $(SQ_1, \ldots, SQ_n)$.
- $MinMVQ$ = sequence number of a data packet where CP_i knows that every active contents peer has known that every contents peer sent every data packet t where $t.SQ \leq MinMVQ$ = min($MinMVQ_1, \ldots, MinMVQ_n$).

$MinMVQ_j$ ($j \neq i$) is the minimum of values M_{j1}, $\ldots$, M_{jn} for a contents peer CP_i, where $M_{jk} = MVQ_{jk}$ if $VW_i.V_k = 1$, $M_{jk} = \top$ otherwise (for $k = 1, \ldots, n$). CP_i knows that every CP_j has transmitted every data packet t where $t.SQ \leq MinMVQ$. "$MVQ_{jk} = \top$" means that CP_j does not perceive CP_k to be active. Here, $No(CP_j)$ shows the order of the contents peer CP_j in $\mathbf{CCP}_i$.

3.2. Data and control packets transmission

A contents peer CP_i transmits data packets to a leaf peer LP_s from the transmission queue XQ_{ik} where k = **Qid**(i, s, n) and n = the number $|\mathbf{CCP}_i|$ of active contents peers. We show the transmission procedure of a contents peer CP_i to send a data packet t to a leaf peer LP_s.

[Transmission procedure of a data packet *t*]
{ k := **Qid**(i, s, n); t := **dequeue**(XQ_{ik});
$SQ_i := t.SQ$; **send**(t, LP_s); }

While transmitting data packets to the leaf peer LP_s, each contents peer CP_i sends control packets with sequence number information $VSQ = \langle SQ_1, \ldots, SQ_n \rangle$ and the view $VW = \langle V_1, \ldots, V_n \rangle$ to all the contents peers. A control packet c sent by CP_i carries the following data:

- $c.VSQ$ = vector $\langle SQ_1, \ldots, SQ_n \rangle$ of sequence numbers, where the ith element SQ_i shows a sequence number of data packet which CP_i has most recently sent to the leaf peer LP_s and each SQ_j (j $\neq$ i) is a sequence number of a data packet where CP_i knows that another CP_j has most recently sent the data packet ($j = 1, \ldots, n$, $j \neq i$).
- $c.VW$ = view $\langle V_1, \ldots, V_n \rangle$ of the contents peer CP_i, where V_j = 1 if CP_i perceives CP_j to be active, otherwise V_j = 0 ($j = 1, \ldots, n$).

Here, $c.SQ_j$ shows an element SQ_j in the vector $c.VSQ$ and $c.V_j$ indicates an element V_j in the bitmap $c.VW$ ($j = 1, \ldots, n$).
[Receipt procedure of a control packet *c*] A contents peer CP_i receives a control packet c from another CP_j:
{ $SQ_j := c.SQ_j$;
MVQ_{jk} := max(MVQ_{jk}, $c.SQ_k$) ($k = 1, \ldots, n$);
$VW_j := c.VW$; $VW_i.V_j$:= 1 if $VW_j.V_j$ = 0; }

[Sending procedure of a control packet *c*] A contents peer CP_i sends a control packet c:
{ $c.VSQ := \langle SQ_1, \ldots, SQ_n \rangle$; $c.VW := VW_i$;
$MVQ_k := SQ_k$ ($k = 1, \ldots, n$);
send(c, CP_j) for every CP_j ($j \neq i$) in $\mathbf{CCP}_i$; }

3.3. View change

By exchanging control packets among the contents peers, each contents peer CP_i detects whether every other contents peer is active or dormant. A control packet c sent by another contents peer CP_j carries the view $c.VW$ of the contents peer CP_j to the contents peer CP_i. The contents peer CP_i has a *consistent* view $\mathbf{CCP}_i$ iff $\mathbf{CCP}_i$ = $\mathbf{CCP}_j$ for every CP_j in $\mathbf{CCP}_i$. Even if another contents peer CP_j perceives another contents peer CP_k to be active, CP_i may perceive CP_k to be dormant since CP_i has not received any control packet from another contents peer CP_k.

[View change] Each time the view $\mathbf{CCP}_i$ changes from inconsistent state to consistent state, the contents peer CP_i changes the transmission procedure as follows:

1. If $VW_i = VW_j$ for every contents peer CP_j such that $VW_i.V_j = 1$, the view $\mathbf{CCP}_i$ is consistent, else stop.
2. $n' := |\mathbf{CCP}_i|$ ($n' \leq n$);
3. Every data packet t where $t.SQ > MinMVQ$ in the packet sequence pkt is redistributed into n' ($< n$) transmission queues $XQ_{i1}, \ldots, XQ_{in'}$. That is, a data packet t is enqueued into the transmission queue XQ_{ih} where h = **Did**($t.SQ$, n').
4. k := **Qid**($No(CP_i)$, s, n');
   t := **dequeue**(XQ_{ik}); **send**(t, LP_s);

[Definition] Suppose there are subsequences $pkt_{1s}\langle t_1]$, $pkt_{2s}\langle t_2]$, $\ldots$, $pkt_{hs}\langle t_h]$ and data packets $t_1, \ldots, t_h$ in a data packet sequence pkt which are sent by the contents peers $CP_1, \ldots, CP_h$, respectively. A collection of the subsequences $pkt_{1s}\langle t_1]$, $\ldots$, $pkt_{hs}\langle t_h]$ are *consistent* iff $pkt_{1s}\langle t_1] \cup \ldots \cup pkt_{hs}\langle t_h]$ is a prefix $pkt\langle t_i]$ where $t_i.SQ$ is the maximum among the data packets $t_1, \ldots, t_h$.

This means, the leaf peer LP_s receives every data packet of a content C which precedes the other packet t_i from the contents peers $CP_1, \ldots, CP_h$. If the packet subsequences $pkt_{1s}\langle t_1]$, $\ldots$, $pkt_{hs}\langle t_h]$ are inconsistent, any contents peer does not send some data packet to LP_s.

[Theorem] Let t_i be a data packet which a contents peer CP_i sends to a leaf peer LP_s. Let t_j be a data packet sent to LP_s by another contents peer CP_j where $t_j.SQ \leq t_i.SQ$ and no data packet t'_j from CP_j where $t_j.SQ < t'_j.SQ \leq t_i.SQ$ ($j = 1, \ldots, n$, $j \neq i$). Here, a collection of subsequences $pkt_{1s}\langle t_1]$, $\ldots$, $pkt_{is}\langle t_i]$, $\ldots$, $pkt_{ns}\langle t_n]$ are *consistent* for every message t_i sent by each content peer CP_i in the asynchronous cooperation algorithm.

Each leaf peer LP_s can receive every data packet from at least one contents peer according to the theorem.

3.4. Dormant contents peers

A dormant contents peer CP_j is detected by each contents peer CP_i as follows:

[Dormant condition] A contents peer CP_i perceives another contents peer CP_j to be *dormant* if $SQ_j <$ max$\{SQ_k \mid VW_i.V_k = 1$, i.e. the contents peer CP_i perceives another contents peer CP_k to be active$\}$ - σ.

The constant σ depends on the number n of contents peers, transmission rate of each contents peer, and delay time between each pair of contents peers. Here, we assume that every contents peer CP_i transmits data packets at a constant rate τ and delay time between every pair of contents peers is a constant δ. The contents peer CP_i perceives a view VW_i of active contents peers by exchanging control packets with other contents peers. Since it takes time to deliver each control packet to every active contents peer and each active contents peer does not send a control packet for some time units, every pair of contents peers CP_i and CP_j may not share the same view, $\mathbf{CCP}_i \neq \mathbf{CCP}_j$. Hence, $\sigma > 2n\delta\tau$. If a contents peer CP_k is detected to be dormant in the contents peer CP_i, $VW_i.V_k$ = 0. If a contents peer CP_k is detected to be active, $VW_i.V_k$ = 1.

Even if a contents peer CP_j is dormant, a leaf peer LP_s can receive the whole content from the other contents peers. If another peer CP_h gets furthermore dormant in a contents peer CP_i, packets in a transmission queue XQ_{ik} where k = **Qid**(h, s, n) are not transmitted to the leaf peer LP_s. Here, packets in n transmission queues $XQ_{i1}, \ldots, XQ_{in}$ are redistributed to ($n - 1$) transmission queues $XQ_{i1}, \ldots, XQ_{i,n-1}$. Here, let n be the number of the active contents peers. Each contents peer CP_i knows that every data packet t where $t.SQ \leq MinMVQ$ is surely transmitted to the leaf peer LP_s. Then, every data packet t where $t.SQ > MinMVQ$ is distributed to one of the n transmission queues by using the function **Did**($t.SQ$, n).

3.5. Redundant transmission

Some contents peer may be dormant. In order to be tolerant of the faults, the contents peers redundantly transmit packets to each leaf peer. For data packets $t_1, \ldots, t_k$, one parity packet pt is created. Even if one data packet t_l is lost, the data packet t_l can be recovered by the other data packets. The number k is decided by the number of active contents peers, i.e. $k \leq n - 1$. One parity packet is inserted every $k - 1$ data packets. Here, let $\boldsymbol{t}_{\mathbf{se}}$ show a parity packet of a subsequence of data packets $t_s, t_{s+1}, \ldots, t_e$ ($s < e$) and e - s + 1 = k in a data packet sequence pkt.

For a subsequence $\langle t_s, t_{s+1}, \ldots, t_e\rangle$ where $e = s + k$ - 1, a parity packet $\boldsymbol{t}_{\mathbf{se}}$ is inserted as follows:

1. if $\mathrm{mod}(s, k) = 1$, $\langle t_s, t_{s+1}, \ldots, t_e, \boldsymbol{t}_{\mathbf{se}}\rangle$.
2. if $\mathrm{mod}(s, k) = 2$, $\langle \boldsymbol{t}_{\mathbf{se}}, t_s, \ldots, t_e\rangle$.
3. if $\mathrm{mod}(s, k) = h$ ($2 \leq h < k$ - 1), $\langle t_s, \ldots, t_{s+h-2}, \boldsymbol{t}_{\mathbf{se}}, t_{s+h-1}, \ldots, t_e\rangle$.

In Figure 4, three contents peers CP_1, CP_2, and CP_3 are transmitting data packets to a leaf peer LP_s. Here, k = 2, i.e. a parity packet is created for two data packets. $\boldsymbol{t}_{\mathbf{12}}$ is a parity packet of two data packets t_1 and t_2. $\langle t_1, t_2, \boldsymbol{t}_{\mathbf{12}}, \boldsymbol{t}_{\mathbf{34}}, t_3, t_4, t_6, \boldsymbol{t}_{\mathbf{56}}, t_5, t_7, t_8, \boldsymbol{t}_{\mathbf{78}}, \ldots\rangle$ is a sequence of data and parity packets. A packet sequence $pkt = \langle t_1, t_2, \ldots\rangle$ is transmitted to the leaf peer LP_s. Here, k = 2. For example, a parity packet $\boldsymbol{t}_{\mathbf{34}}$ is transmitted for a pair of data packets t_3 and t_4. Even if a data packet t_3 is lost by the leaf peer LP_s, the data packet t_3 can be recovered by the data packet t_4 and the parity packet $\boldsymbol{t}_{\mathbf{34}}$. Data and parity packets are transmitted as shown in Figure 4. Even if the contents peer CP_3 is faulty, the leaf peer LP_s can receive the data packet sequence $\langle t_1, t_2, \ldots\rangle$ of the content C because data packets to be sent by the contents peer CP_3 can be obtained from data and parity packets sent by the contents peers CP_1 and CP_2.

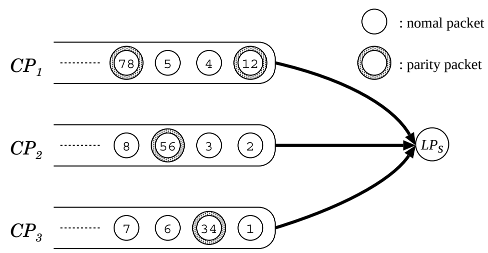


Figure 4. Redundant transmission.

4. Evaluation

We evaluate the asynchronous multi-source streaming (AMSS) protocol. Contents peers $CP_1, \ldots, CP_n$ transmit data packets of a content C to a leaf peer LP. We assume that the delay time among every pair of peers is constant. We consider video data C of one Gbytes as a content C. The video content C is replicated in all the contents peers. Peers are realized on eleven HP BL10e G2 blades of HP Blade Server System with Linux kernel-2.4.26 OS, Intel Pentium M 1.0 GHz CPU and 512 MB main memory where the blades are interconnected in a Fast Ethernet. Each peer is implemented in Java on one blade in the blade server. We emulate delay time and packet loss ratio among each pair of a contents peer and a leaf peer by using NISTnet [2]. Packets are randomly dropped in 20[%] at the leaf peer in this evaluation. We evaluate four types of models, i.e. a single-source streaming model and three multi-source streaming models which are composed of two, five, and ten contents peers.

First, we measure the transmission time of the content and the packet loss ratio for each model as shown in Figures 5 and 6. Here, a video content C of 1 Gbytes is transmitted to the leaf peer LP. The transmission time shows how long it takes for each contents peer to transmit all the data packets. In the figures, the shaded region shows the ratio of data packets lost. In the single contents peer model, it takes 350,000 [msec] to transmit all the packets of the video content C and about 20 % of the data packets are lost due to the buffer overruns and network congetions. On the other hand, in the multi-source streaming models, it takes 180,000 [msec], 107,000 [msec], and 90,000 [msec] to transmit all the data packets of the video content C and about 10 %, 4 %, and 2 % of the data packets are lost, for three models which are composed of two, five, and ten contents peers, respectively. In our approach, the transmission time of multimedia content C can be reduced by more number of contents peers. The bigger packet loss ratio can be reduced if the more number of contents peers are supported.

We measure the throughput for transmitting the multimedia content to a leaf peer LP as shown in Figure 7. Throughput is increased by a asynchronous multi-source streaming (AMSS). Throughput can be increased by more number of contents peers. In the multi-source streaming approach, the performance can be more increased than the traditional single-source streaming model.

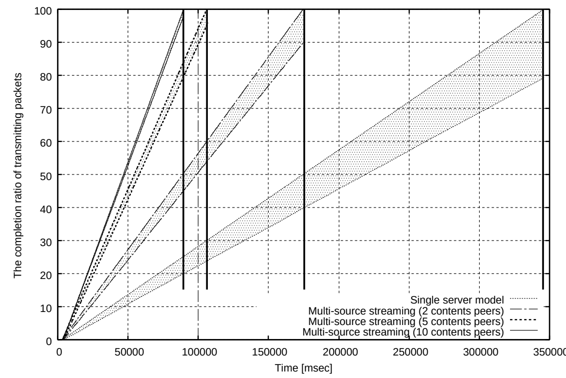


Figure 5. Transmission time.

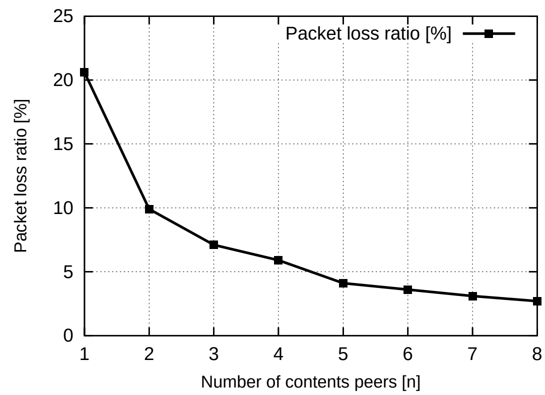


Figure 6. Packet loss ratio.

Next, we measure the jitter delay in the asynchronous multi-source streaming (AMSS) protocol compared with a single-source streaming (SS) protocol. We assume that the delay time among every pair of peers is same and constant. We consider a video data C of 1G bytes as a content. Each peer is implemented in Java. Here, two contents peers are realized in DELL Precision 650 (CPU:dual Intel Xeon 2.0GHz, Memory:1.5GB with Linux kernel-2.4.25) and DELL Precision 530 (CPU:dual Intel Xeon 1.7GHz, Memory:1.5GB with Linux kernel-2.6.10). A leaf peer is immplemented in HP ProLiant DL145 (CPU:dual AMD Opteron 248 2.2 GHz, Memory:2GB with Linux-kernel-2.4.21). The contents peers and leaf peer are interconnected through a routing node HP ProLiant DL145 (CPU:dual AMD Opteron 248 2.2GHz, Memory:2GB with Linux-kernel-2.4.21) in 1 GbE and 10 GbE networks, respectively. We emulate delay time 100 [msec] between a routing node and a leaf peer by using NISTnet in the routing node.

Figure 8 shows the jitter for the AMSS and SS protocols. The AMSS protocol averagely supports 4.6 [msec] jitter while the SS protocol implies the average jitter 20 [msec]. Following the figure, the AMSS protocol can more s atisfy the real-time requirement than the SS protocol.

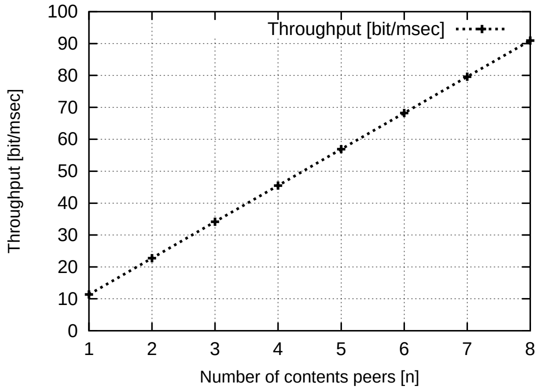


Figure 7. Throughput.

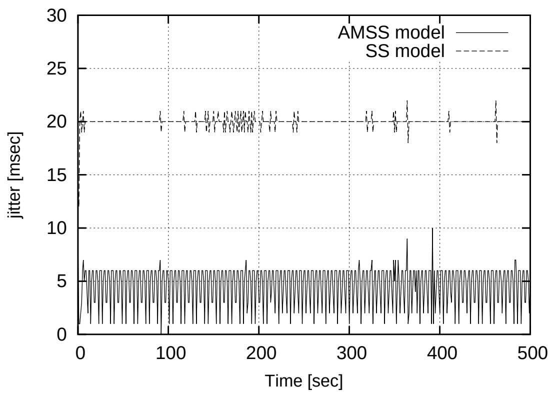


Figure 8. Packet jitter.

5. Concluding Remarks

In this paper, we newly discussed the *asynchronous multi-source streaming* (AMSS) protocol for transmitting continuous multimedia contents from multiple contents peers to a leaf peer. In peer-to-peer (P2P) networks, peers on various types of computers like personal computer support other peers with contents. The peers may not support enough computation power to distribute contents and enough QoS may not be supported in networks. In this paper, we take multi-source streaming approach where multiple contents peers in parallel and minimally redundantly transmit packets of a content to a leaf peer in presence of contents peers faults and packet lost. Here, each contents peer can start transmitting packets independently of the other contents peers. While transmitting packets to leaf peers and exchanging control packets among contents peers, every active contents peer sends a different data packet subsequence of a content from the others to a leaf peer. Even if some number of contents peers get dormant and some packets are lost, each leaf peer can receive the whole content. In the evaluation, we showed that the asynchronous multi-source streaming approach implies high-performance and highly reliable communication than the traditional streaming approach.

In this paper, we assume that every communication channel support the same QoS and each contents peer has the same computation power. We are now discussing a heterogeneous AMSS with heterogeneous contents peers interconnected with heterogeneous channels.

References

[1] 10Gb/s Ethernet. IEEE Standard 802.3ae. 2002.
[2] M. Carson and D. Santay. NIST Net - A Linux-based Network Emulation Tool. *Computer Communication Review*, 33(3):111–126, 2003.
[3] Gigabit Ethernet. IEEE Standard 802.3z. 1998.
[4] S. Itaya, T. Tojo, T. Enokido, M. Rozeta, and M. Takizawa. QoS-Based Synchronous/Asynchronous Data Transmission Model in Group Communication. In *Proc. of the IEEE 18th International Conference on Advanced Information Networking and Applications (AINA-2004)*, volume 1, pages 35–40, 2004.
[5] H. Kamezawa, M. Nakamura, J. Tamatsukuri, N. Aoshima, M. Inaba, and K. Hiraki. Inter-Layer Coordination for Parallel TCP Streams on Long Fat Pipe Networks. In *Proc. of the ACM/IEEE Conference on Super Computing 2004 (SC2004)*, pages 24–33, 2004.
[6] L. Lamport. Time, Clocks, and the Ordering of Events in a Distributed System. *Comm. ACM*, 21(7):558–565, 1978.
[7] A. Nakamura and M. Takizawa. Causally Ordering Broadcast Protocol. In *Proc. of IEEE the 14th International Conference on Distributed Computing Systems (ICDCS-14)*, pages 48–55, 1994.
[8] J. Postel. User Datagram Protocol. *RFC768*, 1980.
[9] J. Postel. Transmission Control Protocol. *RFC793*, 1981.
[10] Project JXTA. http://www.jxta.org/.
[11] P. V. Rangan, H. M. Vin, and S. Ramanathan. Designing an On-Demand Multimedia Service. *IEEE Communications Magazine*, 30(7):56–65, 1992.
[12] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real Time Applications. *RFC1889*, 1996.
[13] H. Schulzrinne, A. Rao, and R. Lanphier. Real-Time Streaming Protocol (RTSP). *RFC2326*, 1998.
[14] K. Taguchi, T. Enokido, and M. Takizawa. Causal Ordered Delivery for Hierarchical Group. In *Proc. of IEEE the 10th International Conference on Parallel and Distributed Systems (ICPADS2004)*, pages 453–460, 2004.
[15] T. Tojo, T. Enokido, and M. Takizawa. Notification-Based QoS Control Protocol for Multimedia Group Communication in High-Speed Networks. In *Proc. of IEEE the 24th International Conference on Distributed Computing Systems (ICDCS-24)*, pages 644–651, 2004.
[16] D. Xu, M. Hefeeda, S. Hambrusch, and B. Bhargava. On Peer-to-Peer Media Streaming. In *Proc. of IEEE the 22nd International Conference on Distributed Computing Systems (ICDCS-22)*, pages 363–371, 2002.

SMIP: Striping Multimedia Communication Protocol for Large Scale Hierarchical Group

Yasutaka Nishimura[1], Naohiro Hayashibara[1], Tomoya Enokido[2], and Makoto Takizawa[1]
[1]*Tokyo Denki University, Japan*
{*yasu, haya, taki*}*@takilab.k.dendai.ac.jp*

[2]*Rissho University, Japan*
eno@ris.ac.jp

Abstract

In traditional hierarchical group protocols, each subgroup communicates with another subgroup through a single gateway communication link. A gateway communication link among subgroups implies performance bottleneck and a single point of failure. In order to increase the throughput and reliability of inter-subgroup communication, messages are in parallel transmitted in a striping way through multiple channels between multiple processes in the subgroups. We discuss a striping multi-channel inter-subgroup communication protocol (SMIP). We evaluate SMIP in terms of stability of bandwidth and message loss ratio.

1. Introduction

Multimedia messages are exchanged among application processes. Each application requires a system to support some quality of service (QoS) like bandwidth, delay time, and packet loss ratio. It is critical to discuss how to support each of huge number and various types of application processes with enough QoS in change of network environments and requirements. In this paper, we discuss how to support flexible group communication service of multimedia data for applications. In peer-to-peer (P2P) [19] and Grid [10] computing systems, hundreds to thousands, possibly million peer processes are cooperating, which are widely distributed in networks.

TCP [21] and RTP [23] support processes with one-to-one and one-to-many transmission of data, respectively. Recently, multiple connections are used to in parallel transmit data from a process to another process in *network striping* technologies [2, 7, 24] in order to increase the throughput. In PSockets [24], data is divided into partitions and each partition is transmitted at a different socket.

In the group communication, processes not only send messages to but also receive messages from multiple processes. Various types of group protocols are discussed to causally deliver messages [17]. In order to reduce the communication overheads, hierarchical groups are discussed. Takamura *et al*. [28] discuss how to support the causally ordered delivery of messages in a hierarchical group by using the vector clock. Here, a group is composed of subgroups where processes in different subgroups exchange messages via gateway processes. Taguchi *et al*. [26, 27] discuss multi-layered group protocols which adopt a vector clock whose size is the number of processes in a subgroup. In Totem [16], messages are ordered by using the token passing mechanism. The protocol cannot be adopted for a large-scale group due to delay time to pass a token. Kawanami *et al.* [13] discuss a hierarchical group where real-time clock is used to causally deliver messages. The authors [18] discuss how to design a hierarchical group from large number of processes by using the *k*-medoid clustering algorithms [12].

In these hierarchical protocols, a gateway process in one subgroup exchanges messages with other subgroups. Each gateway process implies not only performance bottleneck but also single point of failure. In this paper, we discuss a hierarchical group where a pair of subgroups are interconnected through multiple channels among multiple processes in the subgroups to realize parallel, reliable striping communication [24]. That is, a pair of subgroups communicate with one another in a many-to-many communication. In addition, the number of connections among subgroups can be dynamically changed, i.e. the more number of connections are used, the higher bandwidth and reliability are supported for applications.

In section 2, we discuss a model of a hierarchical group. In section 3, we discuss inter-subgroup communication. In section 4, we evaluate the inter-subgroup communication protocol in terms of bandwidth and message loss ratio compared with the one-to-one communication.

2. Striping Hierarchical Group

2.1. Hierarchical group

A *group* of multiple peer processes are cooperating by exchanging messages in order to achieve some objectives. In one-to-one and one-to-many communications [8], each message is *reliably* routed to one and more than one process, respectively. On the other hand, a process sends a message to multiple processes while receiving messages from multiple processes in group communications [4, 5, 7, 17, 25]. Here, a message m_1 *causally precedes* another message m_2 ($m_1 \rightarrow m_2$) if and only if (iff) a sending event of m_1 *happens before* [14] a sending event of m_2 [4]. Here, every process is required to deliver m_1 before m_2. Linear clock [14], vector clock [15], and physical clock with a GPS time server [13] are used to causally deliver messages in distributed systems.

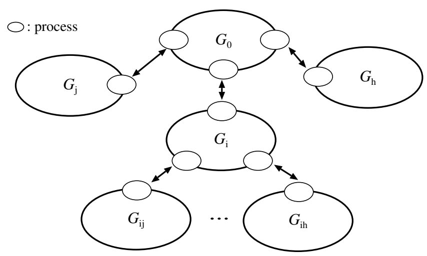


Figure 1. Hierarchical group.

In a *flat* group, every pair of peer processes directly exchange messages with one another. Most group protocols [5,17,25] are discussed for flat groups with the vector clock. Due to computation and communication overheads $O(n)$ to $O(n^2)$ for the total number n of processes in a flat group with the vector clock, a large number n of processes cannot be supported. In addition, it is difficult to maintain the membership in a scalable group. First, processes in a group G are partitioned into multiple subgroups. There is one *root* subgroup G_0 which is connected with subgroups $G_1, \ldots, G_k$ ($k \geq 1$). Then, each subgroup G_i is furthermore connected with subgroups $G_{i1} \ldots G_{ik_i}(k_i \geq 0)$ as shown in Figure 1. Here, G_i is a parent of a child subgroup G_{ij}. In a hierarchical group [26], every pair of a parent subgroup G_i and a child subgroup G_{ij} communicate through one gateway link as shown in Figure 2a. Hence, the gateway processes and inter-gateway communication channel imply performance bottleneck and a single point of failure.

2.2. Inter-subgroup communication

In order to increase the performance and reliability of inter-subgroup communications, we newly discuss a *Striping Multi-channel Inter-subgroup communication Protocol* (*SMIP*). Here, every pair of parent and child subgroups communicate through multiple channels as shown in Figure 2b. A gateway process p_{ij} in G_{ij} communicates with a parent G_i and child G_{ijh}. Gateway processes communicating with G_i and G_{ijh} are *upward* and *downward* gateway processes, respectively, in a subgroup G_{ij}. Each process can be both types of gateways. Normal processes are ones which are not gateways. A *leaf* subgroup includes normal processes and only upward gateway processes. If all the leaf subgroups are at the same layer of the hierarchy, the hierarchical group is *height-balanced*.

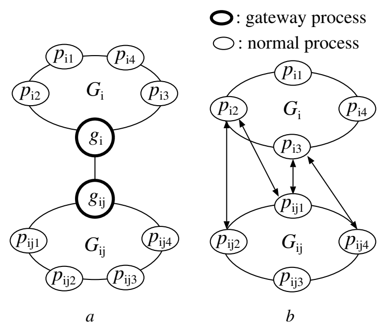


Figure 2. Inter-subgroup communication.

The maximum size of each subgroup is bounded due to the limited computation power of each process. The number s_i of processes in a subgroup G_i has to satisfy $s \leq s_i \leq S$ where s and S show the minimum and maximum numbers of processes in G_i. The smaller size of each subgroup is, the more number of subgroups are connected, i.e. the height or breadth is increased. If the number k_i of child subgroups of G_i is increased, the overhead for inter-group communication is increased. Processes leave and join G_i. In addition, QoS supported by a process or network is changed. Processes in a subgroup may move to another subgroup to satisfy the performance and QoS requirements. If $s_i > S$, G_i is split. If $s_i < s$, G_i is merged into a sibling subgroup G_j, or processes in G_i and its sibling subgroup G_j are redistributed in G_i and G_j. A hierarchical group is dynamically height-balanced as discussed in B-tree [3]. The authors discuss how to construct and maintain a hierarchical group from a large number of peer processes [18].

In this paper, we assume each process broadcasts a message m to all the processes as follows :

1. The process sends m to every process in G_{ij}.
2. An upward gateway forwards m up to downward gateway processes in the parent G_i.
3. Downward gateways forward m down to upward gateways in child subgroups $G_{ij1}, \ldots G_{ijk_{ij}}$.

In each subgroup, a process delivers messages to all the processes by using its own synchronization mechanism like vector clock [15]. Even if a message m causally precedes another message m_2 in a local subgroup, m_1 and m_2 may be causally concurrent in a whole group. In the paper [26], it is discussed how to causally order messages in a whole group.

3. Striping Inter-subgroup Communication

3.1. Inter-subgroup communication

In order to increase the performance and reliability, a pair of parent and child subgroups G_i and G_{ij} communicate with one another through multiple channels with multiple gateways. Here, let us consider a subgroup G_i and its child subgroup G_{ij}. Downward gateway processes in G_i are communicating with upward gateway processes in G_{ij} in a many-to-many communication as shown in Figure 2b.

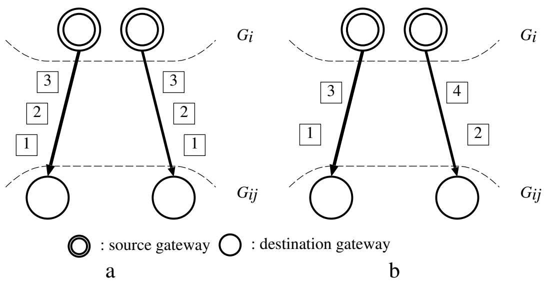


Figure 3. Inter-subgroup communication.

Suppose gateway processes in a subgroup G_i send messages to gateway processes in another subgroup G_{ij}. The former ones are *source* gateways and the latter ones are *destination* gateways in G_i. There are following ways for source gateway processes to send messages to destination gateway processes in G_i [Figure 3] :

a. Each source gateway process sends same messages to the destination gateway processes.
b. Each source gateway process sends messages different from the other gateway processes to destination gateway processes.

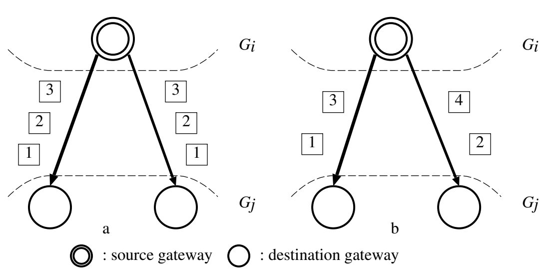


Figure 4. Inter-subgroup communication.

Each source gateway process in G_i transmits messages to multiple destination gateway processes in G_j. There are following ways for each source gateway process to transmit messages [Figure 4] :

a. Same messages are transmitted to each destination gateway process.

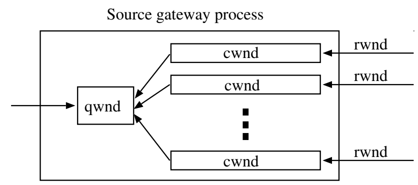


Figure 5. Striping multimedia communication.

b. Different messages are transmitted to each destination gateway process.

If different messages are transmitted in different channels [Figures 3b and 4b], messages arrive at a process out of order. The process has to buffer and then reorder messages. It takes time to reorder messages since a process has to wait for delayed messages. We have to reduce the number of messages to be reordered to increase the performance. We discuss this problem in another paper.

3.2. Striping multi-channel communication

Suppose a gateway process in a subgroup G_i would like to send messages to gateway processes in another subgroup G_j. In this paper, we take the following inter-subgroup transmission protocol from G_i to G_j :

1. One process p_{is} is taken as a source gateway in G_i.
2. On receipt of a message in G_j, the gateway process p_{is} forwards the message to some process, say p_{jt_i} in G_j. Here, p_{jt_1} is a destination process of G_j.
3. On recept of messages in G_i, p_{is} forwards the messages to the destination gateway process p_{jt_i} in G_j.
4. If the channel between a pair of gateways p_{is} and p_{jt_1} might not support enough QoS, the source gateway p_{is} takes another process p_{jt_2} as a gateway in G_j.
5. Thus, the source gateway p_{is} in G_i sends different messages to the destination gateways p_{jt_1} and p_{jt_2} in G_j. p_{is} distributes messages to a pair of gateways p_{jt_1} and p_{jt_2} so that both the channels with p_{jt_1} and p_{jt_2} satisfy the QoS requirement.
6. The larger bandwidth is required, the more number of destination gateways are taken in G_j. The source gateway p_{is} sends messages to the destination gateways in G_i.

Messages are transmitted in a channel between a pair of gateways by the congestion control algorithm used in TCP [11]. If a pair of subgroups are interconnected in a single channel, the subgroups cannot be communicated due to the congestion and fault of the channel. In SMIP, a pair of subgroups G_i and G_j are interconnected with multiple channels. Even if some channel is faulty or does not support QoS requirement, the subgroups can communicate with one another with enough QoS through other operational channels. The network traffic can be distributed

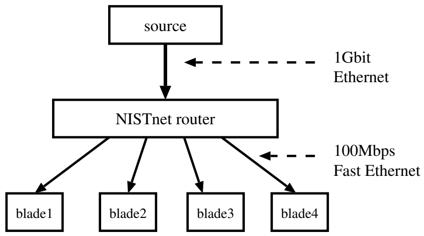


Figure 6. Data transfer arrangement for source striping communication.

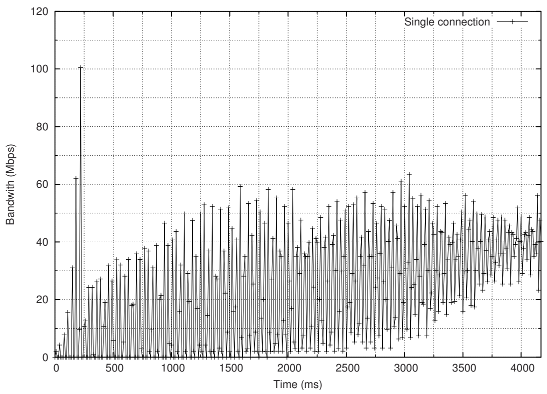


Figure 7. Bandwidth adaptation on traditional one-channel model.

to multiple channels and the other channels compensate the QoS degradation even if QoS of some channel is degraded.

Messages are transmitted in each channel between a pair of source and destination gateways through the congestion control algorithm, *additive increase and multiplicative decrease* (*AIMD*) algorithm used in TCP [11]. Here, two parameters, *congestion window size* (*cwnd*) and *receiver window size* (*rwnd*) are used for each channel. In our protocol, an additional parameter *requirement window* (*qwnd*) showing the size of data in the buffer is used for a set of the channels as shown in Figure 5. The window size (*wnd*) of each channel is decided as $wnd = min(cwnd, rwnd, qwnd)$.

The source gateway p_{is} in a subgroup G_i sends messages to a destination gateway p_{jt_l} in another subgroup G_j through a channel. Then, the window size is calculated. The requirement window size (*qwnd*) is decided as $qwnd = qwnd - wnd$.

4. Evaluation

We evaluate the striping multi-channel inter-subgroup communication protocol (SMIP) in terms of the stability of bandwidth and the message loss ratio compared

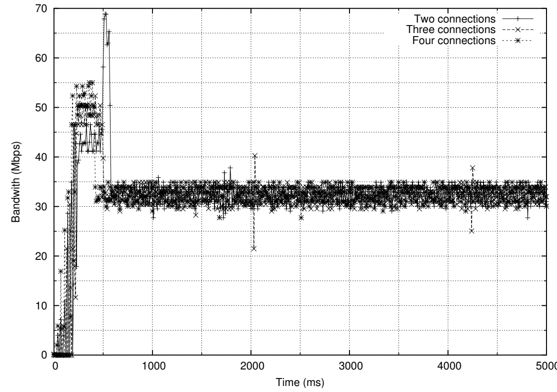


Figure 8. Bandwidth adaptation on striping multi-channel model.

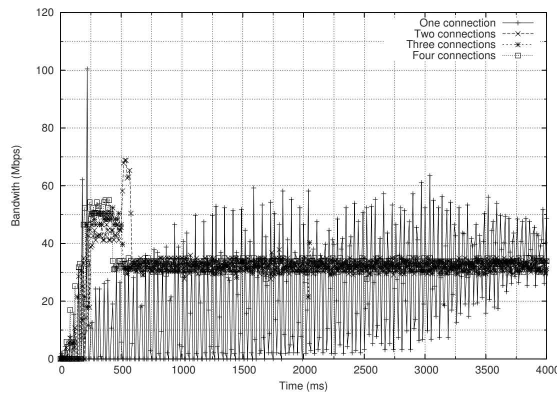


Figure 9. Bandwidth adaptation.

with the traditional one-channel transmission protocol. In the traditional one-to-one communication approach, protocols like RSVP [22] at a lower layer than the transport layer are used to support QoS required by applications. In our striping multi-channel approach, QoS is supported on the end-to-end basis with QoS control at network layer. In the simulation, the bandwidth of the network channel is bounded to be 30Mbps by the evaluation tool although the channel support larger bandwidth 30Mbps means the transmission speed of the digital video (DV) data.

Figure 6 shows the evaluation environment of the striping multi-channel communication protocol. A source gateway process is realized in a computer Dell Precision 530 with dual Intel Pentium Xeon 1.8Ghz and 1.5B memory on Linux 2.6.10. Four destination gateway processes are realized in HP Proliant BL10e blade server with Intel PentiumM 1Ghz and 512MB memory on Linux 2.4.26. These gateway processes are interconnected through a computer HP Proliant DL145 with dual AMD Opteron 2.2Ghz and 2GB memory on Linux 2.4.21 named NISTnet router where NISTnet [6] is installed. The delay time between a source gateway process and a destination gate-

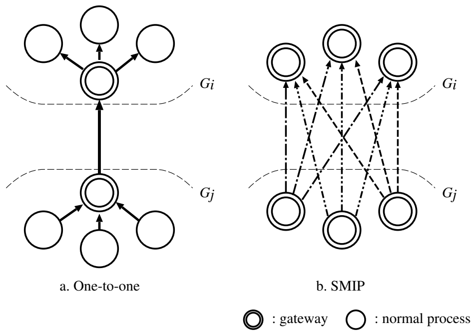


Figure 10. Data transfer arrangement.

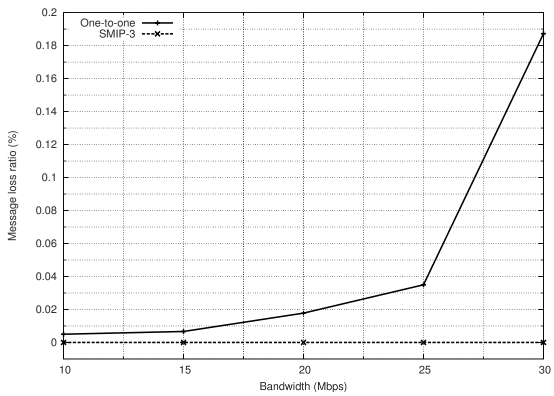


Figure 11. Message loss ratio.

way process is emulated to be 40 milliseconds by using the NISTnet.

In the evaluation, the source gateway process sends multimedia like DV data with 30Mbps. The NewReno algorithm [9] of TCP is used for transmitting messages in each channel. The data transmission procedure of TCP is emulated over UDP/IP [20]. Figure 7 shows how the bandwidth is changed for time in the traditional one-channel transmission. The bandwidth supported is largely changed. Figure 8 shows the bandwidth in our striping multi-channel transmission. Compared with the one-channel transmission, the striping multi-channel transmission supports more stable bandwidth, i.e. 30Mbps. The DV data is required to be transmitted with bandwidth 30Mbps. In the SMIP, the bandwidth of 30Mbps can be continually supported. However, the bandwidth supported by the traditional one-channel protocol is not so stable that the DV data cannot be transmitted. Figure 9 shows both the one-channel and the striping multi-channel ways to show how stable the striping multi-channel way is. Even if QoS is degraded in a channel, messages which cannot be transmitted in the channel can be transmitted through the other channels in the striping multi-channel approach.

Next, we measure the message loss ratio. We take a pair of subgroups G_i and G_j. In the traditional way, one

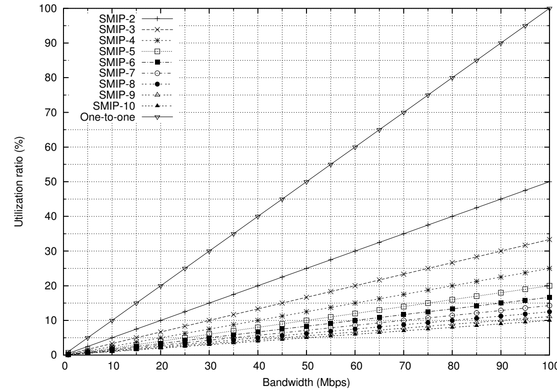


Figure 12. Utilization of bandwidth.

gateway process in G_i communicates with one gateway process in G_j [Figure 10a]. In the SMIP, the same number k of gateway processes in each subgroup of G_i and G_j are interconnected. Here, this inter-subgroup communication from k gateways to k gateways is written in *SMIP-k*. Figure 10b shows SMIP-3. Each pair of gateway processes are interconnected in the 100Mbps Fast Ethernet. Each of normal processes and gateway processes is realized in an HP Proliant BL10e blade server with Intel PentiumM 1Ghz and 512MB memory on Linux 2.4.26. Gateway processes are interconnected through a computer HP Proliant DL145 with dual AMD Opteron 2.2Ghz and 2GB memory on Linux 2.4.21 named NISTnet router where NISTnet [6] is installed. The delay time between G_i and G_j is emulated to be 40 milliseconds by using the NISTnet. Figure 11 shows the packet loss ratio for the bandwidth for each gateway process for the traditional one-to-one and SMIP-3. In the SMIP, no packet is lost. In Figure 11, k [Mbps] means the each of three gateway processes sends packets with $k/3$ [Mbps]. On the other hand, the message loss ratio is increased as the transmission bandwidth of each gateway process is increased. For example, about 0.018% of packets transmitted are lost if a gateway process transmits messages with 60Mbps. Figure 12 shows that the utilization of the bandwidth of each gateway process. In the traditional way, all the bandwidth is used out. This means, the transmission rate cannot be increased. On the other hand, messages are transmitted through multiple channels in SMIP. Hence, each gateway can have unused bandwidth to transmit further messages in SMIP.

5. Concluding Remarks

We discussed the hierarchical group (*HG*) where subgroups are hierarchically interconnected through gateway processes. In order to improve the reliability and throughput of the inter-subgroup communication, a pair of parent and child subgroups are interconnected through multiple communication channels between multiple gateway processes in the subgroup. Gateway processes in different subgroups exchange messages through multiple channels in the striping transmission. In the evaluation, we showed that the striping multi-channel inter-subgroup communication protocol (SMIP) can support the higher stability of the bandwidth and the smaller message loss ratio compared with the traditional protocol.

References

[1] M. Ahamad, M. Raynal, and G. Thia-Kime. An Adaptive Protocol for Implementing Causally Consistent Distributed Services. In *Proc. of the IEEE 18th International Conf. on Distributed Computing Systems (ICDCS 98)*, pages 86–93, 1998.

[2] B. Allcock, J. Bester, J. Bresnahan, A. L. Chervenak, I. Foster, C. Kesselman, S. Meder, V. Nefedova, D. Quesnel, and S. Tuecke. Data Management and Transfer in High-performance Computational Grid Environments. *Parallel Computing Journal*, 28(5):749–771, 2002.

[3] R. Bayer and E. M. McCreight. Organization and Maintenance of Large Ordered Indices. *Acta Informatica*, 1:173–189, 1972.

[4] K. Birman. Lightweight Causal and Atomic Group Multicast. *ACM Trans. on Computer Systems*, pages 272–290, 1991.

[5] K. P. Birman and R. v. Renesse. *Reliable Distributed Computing with the Isis Toolkit*. IEEE Computer Society Press, 1994.

[6] M. Carson and D. Santay. NIST Net: a Linux-based Network Emulation Tool. *Computer Communication Review*, 33(3):111–126, 2003.

[7] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh. SplitStream: High-bandwidth Multicast in a Cooperative Environment. In *Proc. of the 19th ACM Symp. on Operating Systems Principles (SOSP2003)*, pages 298–313, 2003.

[8] S. Deering. Host Groups: A Multicast Extension to the Internet Protocol. *RFC 966*, 1985.

[9] S. Floyd, T. Henderson, and A. Gurtov. The NewReno Modification to TCP's Fast Recovery Algorithm. *Request for Comments 3782*, 2004.

[10] I. Foster and C. Kesselman. *The Grid2: Blueprint for a New Computing Infrastructure*. Morgan Kaufmann Publishers, 2003.

[11] V. Jacobson. Congestion Avoidance and Control. In *Proc. of the ACM Symp. on Communications Architectures and Protocols (SIGCOMM '88)*, pages 314–329, 1988.

[12] L. Kaufman and P. Rousseeuw. *Finding Groups in Data: an Introduction to Cluster Analysis.* John Wiley & Sons, 1990.

[13] S. Kawanami, T. Enokido, and M. Takizawa. Heterogeneous Groups to Causally Ordered Delivery. In *Proc. of the IEEE 6th International Workshop on Multimedia Network Systems and Applications (MNSA 2004)*, pages 70–75, 2004.

[14] L. Lamport. Time, Clocks, and the Ordering of Events in a Distributed System. *Communications of the ACM*, 21(7):558–565, 1978.

[15] F. Mattern. Virtual Time and Global States of Distributed Systems. *Proc. of the International Workshop on Parallel and Distributed Algorithms*, pages 215–226, 1989.

[16] L. E. Moser, P. M. Melliar-Smith, D. A. Agarwal, R. K. Budhia, and C. A. Lingley-Papadopoulos. Totem: A Fault-Tolerant Multicast Group Communication System. *Communications of the ACM*, 39(4):54–63, 1996.

[17] A. Nakamura and M. Takizawa. Causally Ordering Broadcast Protocol. In *Proc. of the IEEE 14th International Conf. on Distributed Computing Systems (ICDCS 94)*, pages 48–55, 1994.

[18] Y. Nishimura, T. Enokido, and M. Takizawa. Design of a Hierarchical Group to Realize a Scalable Group. In *Proc. of the IEEE 19th International Conf. on Advanced Information Networking and Applications (AINA 2005)*, volume 1, pages 9–14, 2005.

[19] A. Oram. *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*. O'Reilly & Associates, 2001.

[20] J. Postel. User Datagram Protocol. *Request for Comments 768*, 1980.

[21] J. Postel. Transmission Control Protocol. *Request for Comments 0793*, 1992.

[22] E. R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification. *Request for Comments 2205*, 1997.

[23] H. Schulzrinne, R. Casner, S. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-time Applications. *Request for Comments 1889*, 1996.

[24] H. Sivakumar, S. Bailey, and R. L. Grossman. PSockets: The Case for Application-level Network Striping for Data Intensive Applications using High Speed Wide Area Networks. In *Proc. of the 2000 ACM/IEEE Conf. on Supercomputing*, page http://www.sc2000.org/proceedings/techpapr/papers/pap.pap240.pdf, 2000.

[25] T. Tachikawa, H. Higaki, and M. Takizawa. Group Communication Protocol for Realtime Applications. In *Proc. of the IEEE 18th International Conf. on Distributed Computing Systems (ICDCS 98)*, pages 40–47, 1998.

[26] K. Taguchi, T. Enokido, and M. Takizawa. Causal Ordered Delivery for a Hierarchical Group. In *Proc. of the IEEE 10th International Conf. on Parallel and Distributed Systems (ICPADS 2004)*, pages 485–492, 2004.

[27] K. Taguchi and M. Takizawa. Two-Layered Protocol for a Large-Scale Group of Processes. In *Proc. of the IEEE 9th International Conf. on Parallel and Distributed Systems (ICPADS 2002)*, pages 171–176, 2002.

[28] M. Takizawa, M. Takamura, and A. Nakamura. Group communication protocol for large group. In *Proc. of the IEEE Conf. on Local Computer Networks (LCN '93)*, pages 310–319, 1993.

Modeling and Analysis of Multipath Video Transport over Lossy Networks Using Packet-Level FEC

Xunqi Yu[1], James W. Modestino[1] and, Ivan V. Bajic[2]

[1]Electrical and Computer Engineering Department, University of Miami
[2]School of Engineering Science, Simon Fraser University

Abstract

The use of forward error correction (FEC) coding is often proposed to combat the effects of network packet losses for error-resilient video transmission on packet-switched networks. On the other hand, path diversity has recently been proposed to improve network transport for both single-description (SD) and multiple-description (MD) coded video. In this work we model and analyze an SD coded video transmission system employing packet-level FEC in combination with path diversity. In particular, we provide a precise analytical approach to evaluating the efficacy of path diversity in reducing the burstiness of network packet-loss processes. We use this approach to quantitatively demonstrate the advantages of path diversity in improving end-to-end video transport performance using packet-level FEC.

1. Introduction

To transmit packet video over lossy packet-switched networks, packet-level forward error correction (FEC) is often proposed to combat packet losses typically due to network congestion, link failures, and timeouts. The efficacy of packet-level FEC is often limited by the bursty nature of typical network packet-loss processes. The use of path diversity, where packets are routed over multiple paths, has recently been proposed to improve network transport for both single-description (SD) and multiple-description (MD) coded video [3, 4]. Path diversity can reduce the effects of extended packet bursts as seen at the FEC decoder, thereby improving the FEC performance. In this work, we model and analyze a SD-coded video transmission system using a combination of both packet-level FEC and path diversity, and demonstrate the efficacy of path diversity in improving joint source-channel coding (JSCC) performance for packet video transmission.

We consider two specific multipath transport scenarios: In the first scenario, we simply assume the paths share no joint links and are totally independent of each other. We provide a precise quantitative analysis of the resulting effective loss-burst-length distribution and residual decoded packet-loss rate using path diversity. Using these results, we demonstrate the efficacy of path diversity in improving end-to-end video transmission performance using packet-level FEC. In the second scenario, we assume different paths may share some joint links and, therefore, the packet-loss processes on different paths may be correlated. We investigate the effect of the resulting path correlation on FEC performance.

The paper is organized as follows: In Section 2 we describe a general system model for packet-video transmission over networks using packet-level FEC and path diversity. In Section 3 we investigate the multipath video transport system for the case of disjoint paths using packet-level FEC. In Section 4 we consider the multipath video transport system for the case of joint paths. Finally, in Section 5 we provide a summary and conclusions.

2. Multipath Video Transport System

Figure 1 illustrates the general SD video multipath transmission system model considered in this paper. Several important system model parameters are also indicated. As shown in this figure, a video transmission system has the following components: a video encoder and decoder, a packet-level FEC encoder and decoder, and a multipath transport network.

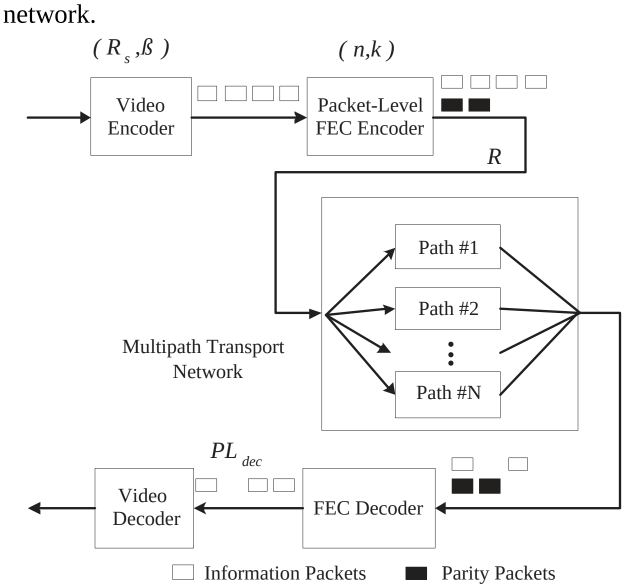


Figure 1. Video transmission system model.

2.1. Video Encoder/Decoder

Assume the source generates a space-time video signal which is used as input to the video encoder. We assume the encoder is a typical block-based hybrid motion-compensated video encoder[1], which encodes the video signal at rate R_s bits/sec with INTRA refresh rate β (as a fraction of macroblocks coded in the intra mode). The latter parameter is indicative of the error resilience capability of the encoded video. We assume the compressed video data are packetized with M macroblocks (MBs) per transport packet.

At the video decoder, the received video packets are depacketized and the video signal is reconstructed. For lost video MBs, passive error concealment will be used to mitigate the distortion due to unrecovered packets.

2.2. Packet-Level FEC Encoder/Decoder

In this paper we use an interlaced Reed-Solomon $RS(n,k)$ coding scheme [6] to provide FEC. For every block of k information packets an additional $p = n - k$ redundant packets are transmitted. The channel-coding rate is then given by

$$R_c = k/n \; ; \qquad \textit{bits/channel use} \tag{1}$$

Assume the total available network bandwidth is fixed at R bps. As a result of the overhead introduced by channel coding, the source coding rate has to be throttled to

$$R_s = R * R_c \text{ bps.} \tag{2}$$

At the FEC decoder, packets received from different paths will be reordered as necessary. Some packets may be lost due to network congestion, link failures, and timeouts. Let $P(j,n)$ denote the block error distribution seen by the FEC decoder after packet reordering, $i.e.$, the probability that j packet losses occur within a block of n consecutive packets, $n \geq 1, 0 \leq j \leq n$. With N_p denoting the number of lost packets within this block, if $N_p > n - k$ we assume the lost packets within this block cannot be recovered by the FEC decoder. Then the residual packet-loss rate of the original video packets after channel decoding can be shown to be given by

$$PL_{dec} = \Big(\sum_{j=n-k+1}^{n} j * P(j,n) \Big) / n \, . \tag{3}$$

2.3. Multipath Transport Network

As illustrated in Fig. 1, the encoded video packets are transported over the multipath transport network composed of N paths. To simplify the analysis, we assume a simple cyclic or round-robin multipath transport scheme: the 1-st packet is transported on path #1, the 2-nd packet is transported on path #2, etc., until the $(N+1)$-th packet which will then be transported on path #1.

[1]In Section 3.3 we will make specific use of the ITU-JVT JM 6.1 codec for the newly developed H.264 video coding standard to provide some numerical examples.

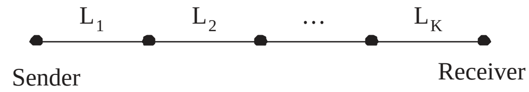


Figure 2. A transport path consisting of K links.

As illustrated in Fig. 2, each path consists of several transmission links connected by routers. Due to temporary buffer overflow and link outages, the packet losses typically occur in bursts with possibly varying burst lengths. We use a two-state discrete-time Markov-chain model, called the Gilbert model [7], to capture the bursty nature of each link. The Gilbert model for link i has two states, "Reception" and "Loss", and two independent parameters: P^i_{01} and P^i_{10}, representing the associated state transition probabilities.

In the literature, two alternative parameters are often used to characterize the Gilbert model: the steady-state packet-loss rate PL^i and the average loss burst length LB^i. The relationships between these parameters are given by:

$$PL^i = \pi^i(1) = \frac{P^i_{01}}{P^i_{10} + P^i_{01}} \; ; \; LB^i = \frac{1}{P^i_{10}} \, , \tag{4}$$

where $\pi^i(1)$ is the steady-state probability of being in the loss state.

We assume the packet-loss processes of the links along a given path are independent. Then, the end-to-end packet loss process of the entire path consisting of K links can be modeled as an aggregate Gilbert channel, with average packet loss rate (PL) and average burst length (LB) given by

$$PL = 1 - \prod_{i=1}^{K} \pi^i(0) \, , \tag{5}$$

and

$$LB = \frac{1 - \prod_{i=1}^{K} \pi^i(0)}{(\prod_{i=1}^{K} \pi^i(0))(1 - \prod_{i=1}^{K}(1 - P^i_{01}))} \, , \tag{6}$$

where $\pi^i(0) = 1 - \pi^i(1)$ is the steady-state probability of being in the reception state for intermediate link i and P^i_{01} is the corresponding transition probability from the reception state to the loss state [7].

2.4. Video Distortion Models

The end-to-end distortion of a reconstructed video sequence, denoted by D, results from two components: the distortion induced by source compression, denoted by D_s, and the channel distortion due to packet losses, denoted by D_c. We make use of the additivity assumption in [1], which states that the end-to-end distortion is the sum of D_s and D_c, $i.e.$,

$$D = D_s + D_c \, . \tag{7}$$

As shown in [1], the compression distortion D_s can be expressed as:

$$D_s = \frac{\theta}{R_s - R_0} + D_0 \, , \tag{8}$$

where R_s is the source coding rate and θ, R_0 and D_0 depend on the INTRA rate β and other model parameters, which are

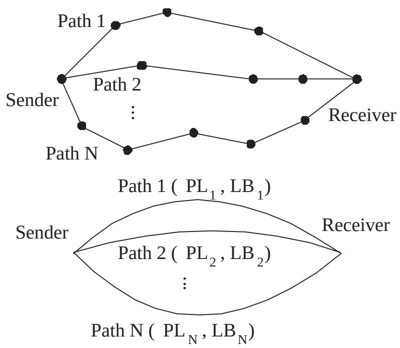


Figure 3. Multipath transport network with disjoint paths.

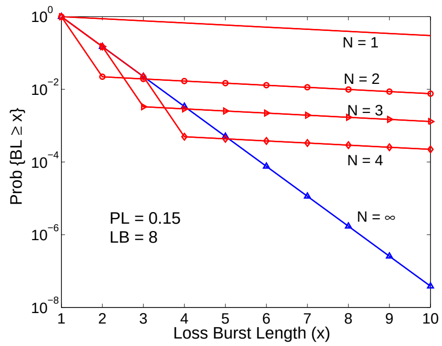


Figure 4. Complementary cdf of loss-burst-length for different N.

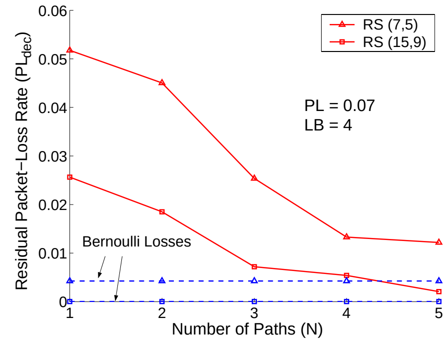


Figure 5. Residual packet-loss rates PL_{dec} vs. number of paths N.

specific to the encoded video sequence and can be obtained by fitting the model to experimental data.

The channel distortion (due to packet losses) D_c can be expressed as a function of the decoded packet-loss rate PL_{dec} and the INTRA coding rate β as [1]

$$D_c = \alpha PL_{dec} \sum_{t=0}^{T-1} \frac{1-\beta t}{1+\gamma t}\,, \tag{9}$$

where $T = 1/\beta$ and the parameter γ describes the efficiency of loop filtering to remove the effects of errors due to packet losses. Likewise, the model parameters γ and α can be obtained by fitting the model to experimental data.

3. Disjoint Paths

First, consider the case where the different paths share no joint links so that packet-loss processes on different paths are independent. Therefore, as illustrated in Fig. 3, each end-to-end path can be modeled as an independent Gilbert channel. The channel parameters associated with the aggregate Gilbert channel for the i-th path, PL_i and LB_i, can be obtained from (5) and (6), respectively[2].

3.1. Analysis of Loss-Burst-Length Statistics

The FEC performance is dependent on the burstiness of the underlying packet-loss processes. Generally, the less bursty the packet losses are, the better performance FEC can achieve. In this subsection, we quantitatively investigate the effectiveness of multipath transport in reducing the burstiness of packet losses. The results are used to explain the FEC performance improvement using path diversity described in the next subsection.

Suppose the random sequence $\{Y_n\}$ represents the packet loss process perceived by an end node, with 1 denoting loss and 0 denoting reception. The effective average loss burst length perceived by an end node can be expressed as

$$ALB_{eff} = \sum_{k=1}^{\infty} k * Pr\{LB_{eff} = k\} = \sum_{k=0}^{\infty} (k+1) * P\{1^k 0|01\}\,, \tag{10}$$

where the random variable LB_{eff} denotes the loss-burst-length seen by the receiver after packet reordering.

Here we model each of the N independent paths as an aggregate Gilbert model. To simplify the analysis, we assume these Gilbert models are homogeneous, each with average packet-loss probability and average burst length PL and LB, respectively. Furthermore, we assume the packets are transmitted over the network using the cyclic multipath transport scheme described in Section 2.3. Then the loss-burst-length distribution, $Pr\{LB_{eff} = k+1\}$, $k \geq 0$, can be expressed as

$$P\{1^k 0|01\} = \begin{cases} PL^k(1-PL) & ; \quad 0 \leq k \leq N-3 \\ PL^{N-2}P_{00} & ; \quad k = N-2 \\ PL^{N-2}P_{01}P_{11}^{k-N+1}P_{10} & ; \quad k \geq N-1\,, \end{cases} \tag{11}$$

where P_{01}, P_{10} are the state transition probabilities of each of the aggregate Gilbert models and can be computed from PL and LB according to (4) with $P_{00} = 1 - P_{01}$, $P_{11} = 1 - P_{10}$.

Therefore, from (10) and (11), the effective average loss burst length is

$$\begin{aligned} ALB_{eff} &= \textstyle\sum_{k=0}^{\infty}(k+1)P\{1^k 0|01\} \\ &= \textstyle\sum_{k=0}^{N-3}(k+1)PL^k(1-PL) \\ &+ \textstyle\sum_{k=N-2}^{N-2}(k+1)PL^{N-2}P_{00} \\ &+ \textstyle\sum_{N-1}^{\infty} PL^{N-2}P_{01}P_{11}^{k-N+1}P_{10} \\ &= 1/(1-PL)\,. \end{aligned} \tag{12}$$

This last expression indicates that, somewhat surprisingly, when the number of paths $N \geq 2$, the effective average loss-burst-length ALB_{eff} is independent of N and LB, but only depends on PL. Figure 4 provides a numerical example of the complementary cdf of the loss-burst-length, $Pr\{LB_{eff} \geq x\}$, with different orders of path diversity N. For comparison, we indicate the corresponding complementary cdf for the Bernoulli channel ($N = \infty$), where the losses are totally independent[3]. It indicates that, although the effective average loss-burst-length ALB_{eff} is the same for $N \geq 2$, the loss-burst-length distributions are quite different. More specifically, it shows that, with an increase of path

[2]For simplicity, we assume each of the disjoint paths are of the same length.

[3]When $N \to \infty$, each packet will be transmitted on a different path. In this case, the channel reduces to a Bernoulli channel, where the packet losses are totally independent.

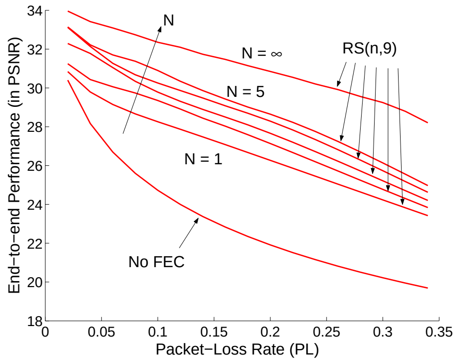


Figure 6. The end-to-end PSNR performance vs. packet-loss rate PL for the Susie sequence with $RS(n, 9)$ codes and path diversity N, where the FEC coding rates R_c are optimally selected; Other model parameters are set as follows: $R = 80$ Kbps, $\beta = 0.02$, $LB = 4$.

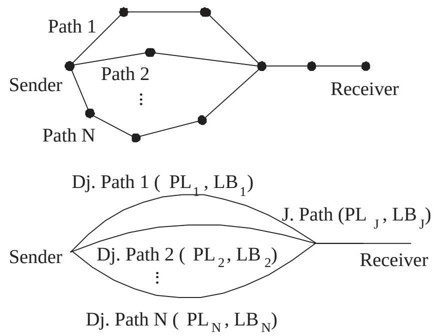


Figure 7. Multipath transport network with joint paths.

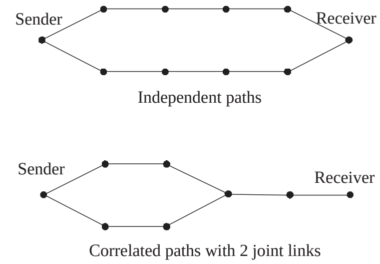


Figure 8. Path correlation model.

diversity order N, the probability $Pr\{LB_{eff} \geq x\}$ will decrease for relatively large x. This means that with increasing path diversity N, the residual packet-loss probability with FEC, PL_{dec}, will be reduced. We will further demonstrate this in the next subsection.

3.2. Analysis of Residual Packet-Loss Rates

We need to obtain the block error distribution $P(j, n)$ to evaluate PL_{dec} according to (3). Consider an arbitrary block of n packets. Without loss of generality, we assume the 1-st packet within this block is transmitted on path #1. Assume there are N paths in total. Then, out of n packets, the number of packets transmitted on the i-th path is

$$l_i = \left\lfloor \frac{n}{N} \right\rfloor + h_i, \qquad 1 \leq i \leq N\,, \tag{13}$$

where

$$h_i = \begin{cases} 1, & \text{if } i \leq n \bmod N\,, \\ 0, & \text{otherwise}\,. \end{cases} \tag{14}$$

Out of these n packets, let b_i denote the number of lost packets transmitted on the i-th path. The total number of lost packets is then

$$\sum_{i=1}^{N} b_i = j\,. \tag{15}$$

Since the packet-loss processes of these N path are independent of each other, we have

$$P(j, n) = \sum_{S} \left(\prod_{i=1}^{N} P_i(b_i, l_i) \right)\,, \tag{16}$$

where the sum is over the set

$$S = \left\{ (b_1, b_2, \ldots, b_N) : \sum_{i=1}^{N} b_i = j \right\}\,, \tag{17}$$

and $P_i(b_i, l_i)$ denotes the block error distribution on the i-th path. If we model the packet-loss process over each path as a Gilbert model, as expressed by (5) and (6), then the block error distribution $P_i(b_i, l_i)$ on each path can be obtained by a recursive algorithm first proposed in [8].

Figure 5 provides a numerical example of the efficacy of path diversity in reducing the residual packet-loss rates for the $RS(7, 5)$ and $RS(15, 9)$ codes, where each path is modeled as a homogeneous and independent Gilbert model with $PL = 0.07$ and $LB = 4$. It demonstrates that, as expected, packet transport with path diversity can improve the efficacy of FEC coding significantly. We have also indicated in Fig. 5 the limiting residual packet loss performance if the losses are independent ($N \to \infty$) [4], $i.e.$, a Bernoulli channel. Observe the rapid approach of the residual packet-loss rates to their limiting values with increasing N. Also note that for use of a fixed code there is little advantage to path diversity orders $N > 4$. Additional results described in [9] also demonstrate the effectiveness of multipath transport in reducing the probability of large bursts and their effect on end-to-end performance.

3.3. Video Performance Using Multipath Transport

Figure 6 demonstrates a comparison of the end-to-end video performance with different network packet-loss rates (PL) for the following cases: 1) without coding ($R_c = 1$) or path diversity ($N = 1$); 2) with an $RS(n, 9)$ code, but no path diversity ($N = 1$), where R_c is optimally selected[5]; 3) combined $RS(n, 9)$ code and path diversity ($N \geq 2$), where R_c is again optimally selected, for the QCIF Susie test sequence. For comparison, we have also indicated the performance achieved on a Bernoulli channel ($N = \infty$). The QCIF Susie sequence (176×144) consists of 150 frames at $f_r = 30$

[4]The limiting value for the $RS(15, 9)$ code is extremely small and indistinguishable from zero in Fig. 5.

[5]Optimally selecting R_c to maximize end-to-end performance for a given overall transmission rate R represents a joint source-channel coding (JSCC) approach.

frames/sec. We assume every $M = 11$ MBs are packetized into one packet. Therefore, each QCIF frame is packetized into $99/M = 9$ packets. Other system parameters are indicated in the figure caption. This figure indicates that, compared to the case of JSCC without path diversity, the combination of JSCC and path diversity can provide significantly improved end-to-end video performance. For example, for $PL = 15\%$, there is a performance advantage of 2 dB in going from $N = 1$ to $N = 4$. However, unlike the fixed code case illustrated in Fig. 5, observe that when the code rate is optimally chosen as part of a JSCC approach there is a considerable performance advantage to path diversity orders $N > 4$.

4. Correlated Paths

In the previous section, we assumed that the different paths share no joint links so that the packet-loss processes on different paths are independent. However, in actual multipath transport networks there may be some shared or joint links between different transport paths. In this case, the packet-loss processes on different paths may be correlated. In actual networks the connection topologies (joint/disjoint links) between the sender and the receiver may be quite varied so that precise modeling of video transport using path diversity can be fairly complex. However, it has been shown in [4, 5] that the important end-to-end properties of a 2-path network can be captured using a simplified three-subpath topology, where subpaths 1 and 2 are formed by the disjoint links along the two paths and subpath 3 is formed by the joint links along both paths. In this section we consider a similar approach as in [4, 5], except that we will consider a more general N-path network. We will specifically concentrate on the effect of path correlation on the end-to-end video performance using packet-level FEC. To model the effect of path correlation on FEC performance, we consider a simplified scenario: different paths share common joint links, as shown in the upper subfigure of Fig. 7, $i.e.$, a number of disjoint links followed by a series of joint links.

4.1. Analysis of Residual Packet-loss Rates

We assume each subpath can be described by a Gilbert model, as illustrated in the bottom subfigure of Fig. 7, where the associated model parameters can be derived from the corresponding portion of the original path using (5) and (6). To evaluate the residual packet-loss rates PL_{dec} after FEC decoding, again we need to determine the block error distribution $P(j, n)$ as seen by the FEC decoder after packet reordering. Assume there are j packets lost out of n consecutive packets. If the total number of packets lost over disjoint links is $b^D = d$ then the number of packets lost over the joint links is $b^J = j - d$. Therefore, we have

$$P(j, n) = \sum_{d=0}^{j} \Big(Pr\{b^D = d\} * Pr\{b^J = j - d\} \Big) . \quad (18)$$

Since out of these n packets d packets have already been lost over the disjoint links, only the remaining $n - d$ packets will be transmitted over the joint links. Therefore, we have

$$Pr\{b^J = j - d\} = P^J(j - d, n - d) , \quad (19)$$

where $P^J(j - d, n - d)$ denotes the block error distribution over the joint path.

Let b_i^D denote the number of packets lost over disjoint path i. Therefore, we have

$$b^D = \sum_{i=0}^{N} b_i^D . \quad (20)$$

Following a similar approach as in going from (15) to (17), it can be shown that [9]

$$P(j, n) = \sum_{d=0}^{j} \Big(\Big(\sum_{S_d} \Big(\prod_{i=1}^{N} P_i^D(b_i^D, l_i) \Big) \Big) * P^J(j - d, n - d) \Big) , \quad (21)$$

where l_i is given by (13) and

$$S_d = \left\{ \left(b_1^D, b_2^D, \ldots, b_N^D\right) : \sum_{i=1}^{N} b_i^D = d \right\} . \quad (22)$$

If we model the packet-loss process over each joint/disjoint subpath as a Gilbert model, then the block error distributions $P_i^D(b_i^D, l_i)$ and $P^J(j - d, n - d)$ on each of the joint/disjoint paths can be obtained by the recursive algorithm originally proposed in [8].

4.2. Effect of Path Correlation on Packet-Level FEC Performance

In order to investigate the effect of path correlation on FEC performance, we consider a simplified 2-path transport scheme and assume there are 5 intermediate links on each of the 2 paths, as illustrated in Fig. 8. We further assume that each intermediate link is independently modeled by a Gilbert channel with parameters PL^i and LB^i. Therefore, the corresponding end-to-end PL and LB can be computed from (5) and (6), respectively, for each path. However, the two paths may share some joint links. Let J denote the number of joint links. The larger J is, the more correlated the two paths are. Figure 8 shows the case of $J = 0$ and $J = 2$. The block packet-loss distribution $P(j, n)$ and the residual packet-loss rate after channel decoding PL_{dec} can be obtained from (21) and (3), respectively.

Table 1 shows the effect of path correlation on the FEC performance with two different RS codes and for two channel conditions. More specifically, it shows the residual packet-loss rates PL_{dec} using a relatively weak $RS(22, 18)$ code and a stronger $RS(31, 18)$ code under two different channel conditions: a relatively bad channel with $PL = 10\%$ and $LB = 8$ and a relatively good channel with $PL = 5\%$ and $LB = 4$. Generally, the results indicate that increased correlation among paths results in a higher residual packet-loss rate PL_{dec} after channel decoding. However, it should be noted that, when the channel conditions are relatively good, or when a weak code is used, the path correlation level (J) does not make much difference in the FEC performance. But when the channel conditions are relatively bad ($PL = 10\%$ and $LB = 8$) and a stronger code is used, the path correlation level J can have a significant effect on the FEC performance. This means that path correlation has a significant impact on FEC performance only when channel conditions are severe and strong FEC codes are used. These analytical results are consistent with the conclusions in [2], where the results there were obtained by simulations.

J	$PL = 10\%, LB = 8$		$PL = 5\%, LB = 4$	
	$RS(22, 18)$	$RS(31, 18)$	$RS(22, 18)$	$RS(31, 18)$
0	8.05 %	3.03 %	2.84 %	0.32 %
1	8.12 %	3.09 %	2.90 %	0.34 %
2	8.20 %	3.34 %	2.95 %	0.39 %
3	8.28 %	3.58 %	3.00 %	0.43 %
4	8.37 %	3.80 %	3.05 %	0.46 %
5	8.46 %	4.01 %	3.11 %	0.49 %

Table 1. Decoded packet-loss rates (PL_{dec}) of a 2-path transport network with different path correlations and FEC code schemes.

4.3. Effect of Path Correlation on Video Performance

Figure 9 demonstrates the effect of path correlation on video performance transmitted over different multipath transport networks. Specifically, it demonstrates a comparison of the end-to-end video performance achieved over multipath transport networks with different numbers of joint links (J) for different path diversity orders (N) and coding strategies (with or without coding) as shown in Fig. 6. The path correlation model is the same as described in Section 4.2, but with different path diversity N. For the coded systems, the code rate R_c is chosen optimally. Other model parameters are indicated in the caption. This figure indicates that, for all path correlation levels (J), increased path diversity combined with JSCC generally can provide improved end-to-end video performance. However, with increased path correlation, the advantage achieved with the use of path diversity will decrease, and with $J \geq 4$ there is little difference in performance independent of the value of N.

5. Summary and Conclusions

We have modeled and analyzed an SD coded video transmission system employing packet-level FEC in combination with path diversity. We provided a precise analytical approach to evaluating the efficacy of path diversity in reducing the burstiness of packet-loss processes. Using this approach we have quantitatively demonstrated the advantages of path diversity in improving end-to-end video transport using packet-level FEC. Finally, we have quantitatively demonstrated the effect of path correlation on the packet-level FEC performance.

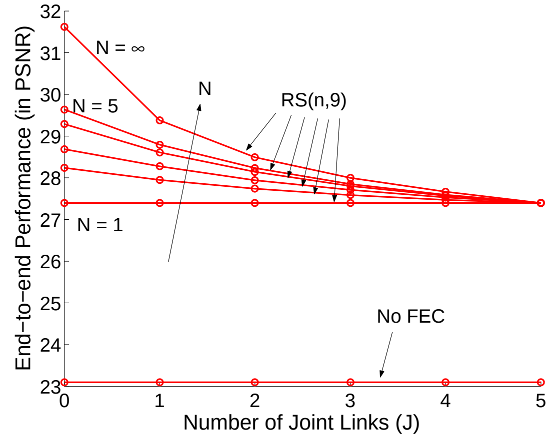


Figure 9. The end-to-end PSNR performance vs. the number of joint links J for the Susie sequence with $RS(n, 9)$ codes and path diversity N, where the FEC coding rates R_c are optimally selected; Other model parameters are set as follows: $R = 80$ Kbps, $\beta = 0.02$, $PL = 15\%$, $LB = 4$.

References

[1] K. Stuhlmuller, *et al.*, "Analysis of video transmission over lossy channels," *IEEE J. Select. Areas in Comm.*, vol.18, pp. 1012-1032, June 2000.

[2] Q. Qu *et al.*, "On the effects of path correlation in multi-path video communications using FEC," *IEEE Globecom*, pp. 977-981, Nov. 2004.

[3] J. G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," *VCIP*, pp. 392-409, San Jose, CA, Jan. 2001.

[4] J. G. Apostolopoulos and M. D. Trott, "Path diversity for enhanced media streaming," *IEEE Communications Magazine*, pp. 80-87, Aug. 2004.

[5] J. G. Apostolopoulos *et al.* "Modeling path diversity for multiple description video communication," *ICASSP*, pp. 2161-64, May 2002.

[6] V. Parthasarathy, J.W. Modestino, and K.S. Vastola, "Reliable transmission of high-quality video over ATM networks," *IEEE Trans. on Image Proc.*, vol.8, pp. 361-374, Mar. 1999.

[7] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Sys. Tech. Journal*, vol.39, pp. 1253-1266, Sept. 1960.

[8] E. O. Elliott, "A model of the switched telephone network for data communications," *Bell Sys. Tech. Journal*, vol.44, pp. 89-109, Jan. 1963.

[9] Xunqi Yu, "Video transmission on packet-switched networks," *Ph.D. dissertation*, Department of Electrical and Computer Engineering, University of Miami, Coral Gables, FL 33124, in progress.

Dynamic Media Routing in Multi-User Home Entertainment Systems

Marco Lohse, Michael Repplinger, and Philipp Slusallek
Computer Graphics Lab, Department of Computer Science
Saarland University, Saarbrücken, Germany
{mlohse, repplinger, slusallek}@cs.uni-sb.de

Abstract

Today, there is a growing interest in home entertainment systems consisting of various networked devices, such as set-top boxes, hi-fi systems, TV sets, or multimedia PCs. However, available solutions only provide a very restricted set of features. Dynamic routing of media streams between distributed devices and multi-user scenarios are typically not supported.

In this paper, we present an architecture for a distributed home entertainment system that overcomes these limitations. We especially examine the most important tasks, namely watching and recording TV. The overall system consists of various devices and allows for multiple users to perform different tasks in parallel. Our architecture supports multi-room applications with a single media stream being presented synchronously on different distributed devices. In addition, applications can be handed over to nearby systems. Finally, we present an algorithm that dynamically determines the optimal media routing for such multi-user scenarios.

1 Introduction

The area of multimedia home entertainment has seen a clear trend towards networked devices, such as set-top boxes, hi-fi systems, or TV sets. Often, existing multimedia PCs are also connected to the home network. In future scenarios, more and more systems will be integrated into our environment invisibly. Besides their general networking capabilities, most of these devices are today fully programmable, which – in principle – allows for advanced application scenarios to be realized.

In contrast, commercially available solutions only provide very restricted functionality. For example, the TV program received by a set-top box can only be rendered to a directly connected display or streamed to a single specialized streaming client. If different media sources are available, e.g. different receivers for analog or digital TV, they can not be integrated seamlessly into the system. Recording TV shows from different sources becomes a tedious task, where each device needs to be programmed manually. More advanced services are not supported.

Previous research in the area of multimedia home entertainment has mainly concentrated on providing location-aware services. Different approaches for accessing media streams using nearby stationary devices are presented in [1] for IEEE 1394 networks and as an OSGi compatible solution in [3]. A solution for "follow-you-and-me video" is described in [7]. Multimedia service delivery with dynamic handoffs to mobile or stationary devices is also presented [2]; synchronized and seamless handoffs were demonstrated in [5]. In [9], an application framework is proposed that can map running applications between different environments depending on the users position.

In contrast, the software architecture presented in this paper provides a transparent view on the network and therefore allows for realizing much more complex scenarios. In particular, we examine the dynamic routing of media streams in multi-user environments, especially for the common tasks of watching and recording TV. Using our system, users can transparently access all available TV channels provided by different distributed sources, e.g. receivers for digital or analog TV connected to the home network. When applications compete for these resources – e.g. when different users try to access the same media source, or a video recording task interferes an already running task – our system tries to share resources or to re-route media streams in order to keep the current quality of service. In addition, a running task can be duplicated to run simultaneously at several locations. As a special case, the media output of an active application can also be handed over between different systems, e.g. to the system that is closest to the user. We argue that such services are essential for future multimedia home entertainment systems.

In the following, we first describe an exemplary home network in Section 2. Then, the various applications and the imposed requirements are pointed out in Section 3. In Section 4, the dynamic routing of media streams will be

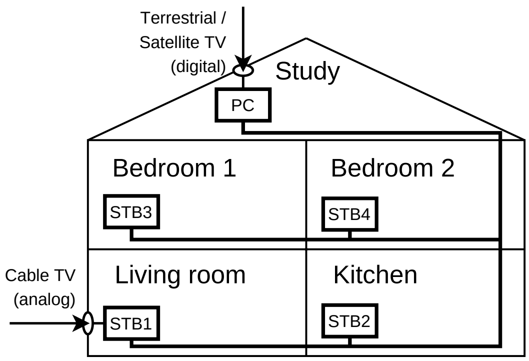


Figure 1. Exemplary setup of a home entertainment network including different TV receivers (analog and digital) and various output devices (STB1-4, and PC).

described for several application scenarios of interest. We start the presentation with relatively simple configurations that only include a single user and one or two running tasks. These configurations will be extended incrementally to include more complex setups. A generic algorithm that optimizes media routing is presented in Section 5. Finally, conclusions are drawn and future work is presented in Section 6.

2 Exemplary Home Network

Figure 1 shows an exemplary setup of a network for multimedia home entertainment. Notice that we use a moderate sized example for simplicity; the presented approach can also be applied for larger numbers of users and devices. Our setup includes different devices connected using commodity fully switched 100 Mbit LAN networking technology that is able to transmit several encoded high quality audio/video streams in parallel.

Two sources for receiving TV are available in our example. The set-top box in the living room (STB1) receives a large number of channels via analog cable TV and is used to watch TV using a connected display. Furthermore, we assume that this device also includes a hardware unit for encoding media streams to MPEG2 in real-time. Such encoded streams can then either be stored on the hard disc integrated into the device itself or transmitted to other devices within the network.

As second source, a PC in the study contains a board for receiving terrestrial or satellite TV, e.g. using one of the standards defined by Digital Video Broadcasting (DVB), Advanced Television Committee (ATSC), or Integrated Services Digital Broadcasting (ISDB). Since the received streams are already available as MPEG2, they can either be stored on the hard disc of the PC or forwarded to other devices.

In particular, three additional set-top boxes, STB2, STB3 and STB4, are available within the kitchen and the two bedrooms, respectively. These devices can be used to decode and render MPEG2 streams received via the network. Notice that this feature is also supported by STB1 and the PC.

We assume that all these devices are fully programmable. In particular, each device runs a software architecture supporting distributed multimedia – the Network-Integrated Multimedia Middleware (NMM) [4]. The features of this architecture will be described in more detail in Section 4.

Notice that TV sources provide different quality. For example, we assume that streams received digitally (PC) offer a better quality then analog TV encoded to MPEG2 (STB1). Furthermore, each TV source provides a specific list of channels. For the following examples, we assume that the PC includes a newly purchased DVB-T board for receiving terrestrial digital TV – a typical scenario for Germany and other European countries. Therefore, only 15-20 TV channels are available using the digital TV receiver, compared to 30 or more for analog TV. While some channels might be available for both sources, some can only be watched by accessing a specific device. To take the differences in quality and number of channels into account, we use a *unified channel list*. Within this list, all available channels (for analog and digital TV) are arranged according to users' preferences. For each channel, available TV sources and the corresponding devices are given and sorted according to their "quality", i.e. digital TV is preferred over analog.

3 Applications and Requirements

For the two users present in our setup, namely Alice and Bob, following applications are provided.

- *Live TV:* A user watches a specific channel using a TV source (analog using STB1 or digital using PC) and an output device (STB1, STB2, STB3, STB4, or PC). Channel-hopping is fully supported, i.e. a user can switch between channels received by different devices seamlessly. Live TV supports multi-room playback, i.e. the same media stream can be rendered synchronously in several rooms using different available devices. A special case thereof is a "follow-me" scenario, where a user can hand over the media output of an already started live TV application to his current location, e.g. when moving between rooms.

- *Video recorder:* Records a TV show encoded as MPEG2 to the hard disc of STB1 or the PC, respectively. A recording is specified by a *timer* that includes start and end time an the intended channel; the TV source to be used is determined when programming the timer.

- *Electronic Program Guide (EPG):* A single EPG task (e.g. running on the PC) collects program information transmitted within the digital data stream and stores it within a database. All devices can access this database for displaying additional information on the current TV show. Furthermore, users can browse the EPG database for easily programming timers for video recorders.

Notice that the system described in the following is able to handle several simultaneously running instances of the live TV and the recorder application, e.g. because multiple users are active or more than a single TV show is to be recorded at a time.

From the above described applications, following general requirements can be derived.

- Access, control, and integration of media streams received by distributed devices needs to be provided, e.g. for remote TV receivers connected to the network.

- Simultaneous and shared access to a single distributed device has to be supported, e.g. to allow for different users to watch the same TV channel using different output devices in different rooms.

- Synchronous playback of media using several networked systems is required, e.g. to avoid offsets in the audio and video presentation when a single stream is to be rendered using output devices located in adjacent rooms. This feature is needed for supporting multi-room and "follow-me" presentations.

- A last requirement to be fulfilled is the appropriate handling of the dynamic behavior of the overall system including concurrently operating applications (e.g. live TV and channel-hopping, video recorders, or background tasks that acquire EPG information) competing for shared resources (e.g. TV receivers).

4 Dynamic Media Routing

We first describe the architectural approach of the underlying multimedia framework used for our system. Then, the dynamic routing of media streams is presented for different use cases of importance for home entertainment.

4.1 Architectural Model

Within common multimedia architectures, such as DirectShow [8] or the Java Media Framework [10], all multimedia functionality is modeled by *flow graphs*. A flow

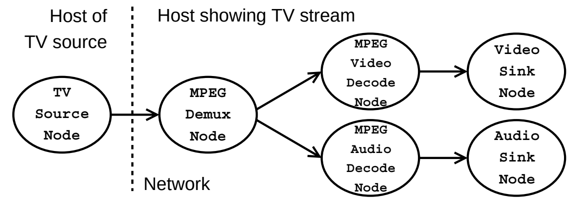


Figure 2. Flow graph for live TV including a possibly remote TV source and nodes for demultiplexing, decoding and rendering the received MPEG2 audio/video stream.

graph is a directed graph consisting of independent processing elements. Multimedia data streams are encapsulated into buffers that are forwarded in "downstream" direction from sources of data to sinks. Intermediate nodes perform transformations on data streams, e.g. decoding or demultiplexing.

While flow graphs of commonly available architectures are restricted to operate on a single host, the underlying NMM framework used for our work allows for transparently *distributed flow graphs* [4]. Processing elements, called *nodes*, running on remote systems can be connected and controlled the same way as locally running elements. A *registry service* supports the search for specific components within the network and handles the creation of distributed flow graphs.

4.2 Live TV and Channel-Hopping

Since NMM allows for transparently distributed flow graphs, the above described live TV application can be realized by creating the setup shown in Figure 2: a flow graph including a possibly remote TV source and nodes for demultiplexing, decoding and rendering the received MPEG2 audio/video stream. Notice that any device in the home network can host this application and therefore be used as output device. If, for example, user Alice wants to watch TV in the kitchen using STB2, the corresponding application will request a TV source from either STB1 or the PC; the audio/video presentation will performed on STB2.

The TV source to be used is determined by a look-up in the unified channel list (compare Section 2). When Alice changes channels, the system checks if the newly selected channel is still supported by the currently used TV receiver. If not, the flow graph is dynamically adapted, i.e. the source node is replaced. To avoid interrupting the presentation, the new data source is first requested and configured for the selected channel before the data stream from the currently used source node is interrupted. Together, this allows for seamlessly and transparently switching between different distributed TV receivers.

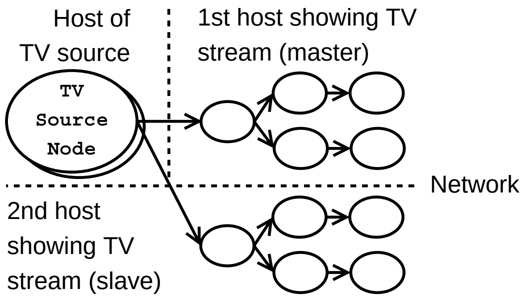


Figure 3. The source node for receiving TV is shared among two independent distributed flow graphs to enable multi-room live TV and a follow-me feature.

4.3 Multi-Room Live TV and Follow-Me TV

Another important use case occurs if Alice wants to continue watching TV while moving within the home. As an example, consider the situation when Alice wants to watch TV while preparing a meal in the kitchen. Then, she changes her location several times, e.g. she enters and leaves the living room when serving dinner. Such scenarios require to route the same media stream to several output devices simultaneously.

Within our architecture, a special service called *session sharing* provides this functionality [6]. This service automatically maps request for flow graphs to be created newly to already running flow graphs. All nodes within running flow graphs that are "identical" are determined and then shared among different applications. Conceptually, this results in two independent applications, each using a specific flow graph that internally shares certain nodes with another flow graph.

For our example, the result of this process is shown in Figure 3. The source node for receiving TV is shared among two applications operating on different hosts. Since Alice started to watch TV in the kitchen using STB2, the corresponding flow graph is system is called *master*; the additionally created flow graph running on STB1 in the living room is called *slave*. Notice that full control is provided for both systems. This means that Alice can change channels operating STB1 or STB2, e.g. using a remote control.

In addition, the underlying NMM architecture provides lip-synchronous playback of audio and video on all participating devices. The offset between different systems is reduced to only a few milliseconds, which is sufficient for multi-room solutions [6]. This is achieved by using following approach. First, the internal clocks of all systems are synchronized using the Network Time Protocol (NTP). Second, the maximum networking and processing delay of all paths within all shared flow graphs is determined. The audio and video presentation of all devices is then delayed artificially until this offset is reached. During runtime, the maximum delay is updated if necessary.

For realizing a "follow-me" service, the session sharing service is first used to create an additional shared flow graph that renders audio and video on a different system, e.g. a nearby system that is manually selected by the user. Once the slave graph is running, the master graph is destroyed.

4.4 Live TV versus EPG

In this use case, Bob starts watching sports using the TV in the living room connected to STB1. Let us assume that the live TV application chooses the digital TV receiver (PC) because it is the preferred data source for the selected channel. However, since the EPG application is continuously running in the background and using the digital receiver to collect EPG information, we define following rule:

> *Live TV can take control of a TV source that is used by an EPG task.*

When the live TV application running on STB1 requests the digital TV source, only a shared instance is returned by the session sharing service as described in Section 4.3. Based on our rule, the live TV application takes control of this source and informs the EPG task that channel switching is no longer allowed. Therefore, the EPG task tries to use another digital TV source. Since there is no such resource available in our setup, the EPG task continues to collect program information from the channel that is currently watched by the live TV application controlled by Bob. As soon as the live TV application controlled by Bob stops using the digital TV receiver, the EPG task takes over control again and continues collecting program information from all channels.

4.5 Video Recorder Timers

In the next use case, Bob wants to schedule the recording of a TV show to be broadcasted next week. Therefore, he selects the corresponding entry using the EPG, which in turn creates a timer to be stored in a global database accessible by all recording tasks. If there are no other conflicting timers, the digital TV receiver is chosen as preferred data source.

However, in general, there will be several timers provided by different users. Therefore, whenever a new timer is created, the EPG checks whether it can schedule all recordings using all available TV sources. For assigning TV sources to timers, we use a depth-first search. The algorithm starts by assigning the most preferred TV source for the specified channel to the timer that was provided first. Then, this process is repeated for all further timers. In each step, the most preferred TV sources are checked first. As soon as a TV sources is assigned to each timer, the algorithm terminates. If no solution can be found at all, the last timer provided is discarded and a message is displayed to inform the user that the timer could not be set.

In the worst case, the algorithm needs to test n^t combinations, where n is the number of TV sources and t the number of timers. However, since we use a greedy depth-first search that terminates once a solution is found, the runtime of the algorithm is acceptable even for larger numbers of TV sources and timers. For future work, we would like to apply more advanced approaches.

4.6 Multi-User Live TV

Another use case occurs if several users are watching TV concurrently and try to access the same TV source, e.g. during channel-hopping. For handling this case, we define following rule:

> *The live TV application that requests a TV source first obtains full control of it.*

As an example, consider that Bob is still watching sports using the TV in the living room connected to STB1; the corresponding live TV application is using the digital TV receiver as data source (PC). If Alice starts watching TV in the kitchen using STB2, she will only obtain full control of the analog TV source – the digital TV source can only be shared without any control options. When Alice switches to a channel that is preferably received via digital TV (or that is only available using the digital TV receiver), the live TV application controlled by Alice requests the currently set channel of the digital TV source *before* using it as shared node within its flow graph (compare Section 4.2). If the channel requested by Alice is not the channel that Bob is currently watching, another TV source is tried to be chosen, e.g. the analog TV source. If the channel is not available using that source, a message is displayed, telling Alice that the chosen channel is currently not available since the required TV source is controlled by Bob.

4.7 Multi-User Live TV versus Recording

Finally, the most demanding use case occurs if several users are watching live TV and a video recording must be started. According to traditional analog video cassette recorders, our recording application will switch to the specified channel of the assigned TV source even if a user is currently watching a different channel using that source. Together, we apply following rule:

> *A video recorder can take full control of a TV source that is used by another task, such live TV or the EPG.*

Let us extend the example presented in Section 4.6. Again, Alice and Bob are watching TV concurrently. The timer to be recorded next is scheduled to use the digital TV source (compare Section 4.5). Thus, the video recorder requests the digital TV source, but only obtains a slave graph since this device is already used by Bob. Based on our rule, the video recorder takes control of the TV source, switches to the intended channel and starts recording – in fact, it becomes the master. Therefore, Bob is notified about this change.

If the video recorder accesses the same program that Bob is currently watching, the running live TV application is continued without further modification. Otherwise, the live TV application tries to find another TV source that can be used for watching the channel that was originally selected by Bob. Since no such TV source is available in our example, a message is displayed stating that the channel is no longer available. In this case, Bob can either watch the channel that is currently recorded or manually cancel the recorder application. In addition, Bob can switch to the channel that Alice is currently watching. Alice still has full control of the analog TV source and can switch between all its available channels.

5 Generic Media Routing Algorithm

Based on the previous sections we derive a generic algorithm for media routing. The algorithm expects three arguments. The first, `source_list`, is a list that includes all TV sources that provide the channel to be selected, which is given as second argument called `channel`. The third argument `application` specifies the application that invoked the algorithm.

The algorithm is split into three parts. In the first part, we try to find a TV source that can be fully controlled by the application. The main idea is to request each suitable TV source using the session sharing service (`requestSource()`). This service returns a master graph if the TV source is currently unused. Then, we specify the application to be the "master" of the TV source (`setMaster()`), i.e. the TV source will only accept commands, like channel switching, from the application specified.

Otherwise, the algorithm tries to get full control of the TV source based on the rules defined in Section 4. If our application can take control of a TV source, we notify the original master of this source that it can no longer switch the channel (`removeMaster()`). Since the original master of the TV source may need another TV source to continue its work, it can also invoke this algorithm to find another possible solution.

If no possibilty was found to acquire full control of a TV source, a TV source that is already set to the wanted channel is tried to be used as shared node (`currentChannel()`) within the second part of the algorithm.

Otherwise, the third part of the algorithm is reached: An error message is returned, if the method was invoked by a live TV application, because no suitable TV source can be used. If the algorithm was invoked by an EPG task, we simply return the first TV source of the list of TV sources that can be used to collect at least the program information of the corresponding selected channel. Together, the pseudo code of the algorithm is as follows.

```
tv_source searchTVSource( source_list , channel , application )
{

  /* First part of the algorithm */
  foreach( tv_source in source_list ) do {

    if( requestSource( tv_source ) == Master ) {
      setMaster( tv_source , application );
      return tv_source;
    }

    if( application == LiveTV ) {
      if( isMasterOf( tv_source ) == EPG ) {
        removeMaster( tv_source );
        setMaster( tv_source , application );
        return tv_source;
      }
    }

    if( application == VideoRecorder ) {
      if( isMasterOf( tv_source ) == EPG or
          isMasterOf( tv_source ) == LiveTV ) {
        removeMaster( tv_source );
        setMaster( tv_source );
        return tv_source;
      }
    }

  }

  /* Second part of the algorithm */
  foreach( tv_card in source_list ) do {

    if( currentChannel( tv_card ) == channel) {
      return tv_card;
    }

  }

  /* Third part of the algorithm */
  if( application == LiveTV ) {
    throw Exception( ``Channel is currently not available'' );
  }

  if( application == EPG ) {
    return firstEPGSource( source_list );
  }

}
```

6 Conclusions and Future Work

In this paper, we presented an architecture for dynamically routing media streams in multimedia home networks. While we especially examined the important tasks of watching and recording TV, the described approach can be applied to other sources of media as well. The overall system consists of various networked devices, such as set-top boxes and PCs. Different types of TV sources are seamlessly integrated and can transparently accessed by all applications. Users are allowed to watch TV on any device, media sources are determined automatically. Furthermore, our architecture supports multi-room and location-aware presentations. When different applications, such as live TV, video recording, or an EPG task, are operating concurrently, a generic algorithm determines an optimal routing of media streams and dynamically adapts to changing conditions.

Future research will concentrate on the integration of mobile devices. Especially the automatic adaptation of media streams for such resource-poor systems seems to be demanding. Since users currently have to select their location manually, we would also like to include solutions for tracking users and mobile devices into our architecture.

References

[1] H. Baldus, M. Baumeister, H. Eggenhuissen, A. Montvay, and W. Stut. WWICE – An Architecture for In-Home Digital Networks. In *Proceedings of Multimedia Computing and Networking (MMCN)*, 2000.

[2] Y. Cui, K. Nahrstedt, and D. Xu. Seamless User-level Handoff in Ubiquitous Multimedia Service Delivery. *Multimedia Tools and Applications Journal, Special Issue on Mobile Multimedia and Communications and m-Commerce*, 22, 2004.

[3] H.-J. Eikerling and F. Berger. Design of OSGi Compatible Middleware Components for Mobile Multimedia Applications. In *Protocols and Systems for Interactive Distributed Multimedia Systems, Joint International Workshops on Interactive Distributed Multimedia Systems and Protocols for Multimedia Systems, IDMS/PROMS 2002, Proceedings*, 2002.

[4] M. Lohse, M. Repplinger, and P. Slusallek. An Open Middleware Architecture for Network-Integrated Multimedia. In *Protocols and Systems for Interactive Distributed Multimedia Systems, Joint International Workshops on Interactive Distributed Multimedia Systems and Protocols for Multimedia Systems, IDMS/PROMS 2002, Proceedings*, 2002.

[5] M. Lohse, M. Repplinger, and P. Slusallek. Dynamic Distributed Multimedia: Seamless Sharing and Reconfiguration of Multimedia Flow Graphs. In *Proceedings of the 2nd International Conference on Mobile and Ubiquitous Multimedia (MUM 2003)*, 2003.

[6] M. Lohse, M. Repplinger, and P. Slusallek. Session Sharing as Middleware Service for Distributed Multimedia Applications. In *Interactive Multimedia on Next Generation Networks, Proceedings of First International Workshop on Multimedia Interactive Protocols and Systems, MIPS 2003*, 2003.

[7] J. Nakazawa and H. Tokuda. A Pluggable Service-to-Service Communication Mechanism for Home Multimedia Networks. In *Proceedings of ACM International Conference on Multimedia*, 2002.

[8] M. D. Pesce. *Programming Microsoft DirectShow for Digital Video and Television*. Microsoft Press, 2003.

[9] M. Roman, C. K. Hess, R. Cerqueira, A. Ranganathan, R. H. Campbell, and K. Nahrstedt. Gaia: A Middleware Infrastructure to Enable Active Spaces. *IEEE Pervasive Computing*, 1(4), 2002.

[10] Sun Microsystems. *Java Media Framework API Guide*, JMF 2.0 FCS edition, 1999.

Proceedings

The 2005 International Workshop on Visual Languages and Computing (VLC 2005)

Co-Editors

Philip Cox, Dalhousie University, Canada
Trevor Smedley, Dalhousie University, Canada

Tableaux for Diagrammatic Reasoning

Octavian Patrascoiu[1], Simon Thompson [1], and Peter Rodgers[1]
[1] *Computing Laboratory, University of Kent, United Kingdom*
{O.Patrascoiu, S.J.Thompson, P.J.Rodgers}@kent.ac.uk
http://www.cs.kent.ac.uk/rwd/

Abstract

Diagrammatic notations, such as the Unified Modeling Language (UML), are in common use in software development. They allow many aspects of software systems to be described diagrammatically, but typically they rely on textual notations for logical constraints. In contrast, spider diagrams provide a visual notation for expressing a natural class of set-theoretic statements in a diagrammatic form. In this paper we present a tableau system for spider diagrams, and describe an implementation of the system. In a software development context, the system allows users to explore the implications of design choices, and thus to validate specifications; beyond this, the tableau algorithm and system are of general interest to visual reasoners.

1. Introduction

Tableaux provide an intuitive mechanism for exploring the models and counter-models of logical formulas, and in particular they give mechanisms for deciding satisfaction and validity for a wide class of logics. To users, tableaux are of value not only as decision procedures but also by providing a mechanism by which a user can explore the consequences of a statement or set of statements.

This is particularly important when a statement is used as the specification of or a constraint on a software system. In software development, it is a well-known problem that specifications can suffer from incompleteness, inconsistency, or inappropriateness to the problem domain. It is therefore crucial that specification writers have the chance to engage and interact with their specifications in as many different ways as possible.

Obviously specifications should be checked for syntactic and type correctness, and this can be done in a routine way. In order to understand the semantics of the formulas, other mechanisms are needed. A decision procedure will allow a user to find out whether a specification is satisfiable, but this does not answer the question of whether the intention of the specifier has been realized. To achieve this it is necessary to tease out the significance of the formula, and specifically

- to investigate the possible models of the formula, and
- to explore the consequences of the formula: in other words, to discover its ‘implications’.

Tableaux can provide both of these for the language of spider diagrams. A spider diagram gives a diagrammatic representation of a statement about a finite number of sets, their membership and their interrelationships. For instance, in the context of specification such diagrams can be used to describe the relations between objects and classes.

The language of spider diagrams is equivalent to monadic predicate logic with equality [18]. It would therefore be possible to turn diagrammatic representations into textual statements and to apply decision procedures or tableau methods to the translations of diagrams. This would be perfectly adequate in the case of a decision procedure, but where feedback to the user is necessary – about the form of models, or the consequences of a formula, say – then it is crucial to work with a visual representation in order to provide recognizable visual feedback. Hence the system developed in this paper.

The paper begins in Section 2 with an overview of diagrammatic reasoning, tableaux and spider diagrams and their reasoning rules. Section 3 presents the central tableau algorithm for spider diagrams, illustrates it by a number of examples, discusses heuristics and optimizations and concludes by evaluating the system

The conclusion reviews the work presented in the paper, and explores prospects for future work.

2. Spider Diagrams

The motivation for spider diagrams comes from the belief that visual representations of logical statements can aid understanding of the underlying meaning, and are more acceptable to people who are unfamiliar with standard textual mathematical notations. A further reason is that many essentially visual systems have to resort to textual notation for indicating logical expressions over visual diagrams. An example is the Unified Modeling Language (UML) [19], which is used for describing the design of object-oriented systems. UML is entirely diagrammatic, except for the language used to describe complex constraints on collections of objects: this is the Object Constraint Language (OCL) [12]. Researchers in diagrammatic reasoning are developing candidates for replacing OCL with a visual notation; spider diagrams are one such example.

2.1. Background

The work described here is performed as part of the Reasoning with Diagrams project [15], which engaged in developing spider diagrams and similar diagrammatic reasoning methods. Spider diagrams are an extension to work of Shin [16]. Shin presented formal systems of Venn-Peirce diagrams: Venn diagrams extended with annotations to indicate empty and non-empty sets. Venn-Peirce diagrams admit purely diagrammatic reasoning and Shin proved that they could be equipped with sets of logical rules that are both sound and complete.

In related work, Hammer [6] presented a sound and complete system of Euler diagrams [3]. Sound and complete sets of diagrammatic inference rules have also been developed for several systems of spider diagrams [7].

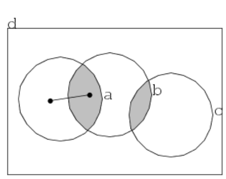


Figure 1 A spider diagram

Spider diagrams [5] are themselves a subset of the constraint diagram notation [8]. Spider diagrams represent the interrelationships and membership of a finite collection of sets. In Figure 1 diagram **d** contains three sets **a**, **b** and **c** which are represented by *contours* (simple closed plane curves). *Regions* are given by intersection, union and complementation, and a *zone* is a region which properly contains no other regions.

The figure contains regions corresponding to **a∩b**, **a∪b** and so forth but not to **a∩c**, say; **a∩b** is a zone, but **a∪b** is not, since it properly contains the zone representing **a∩b** (amongst others). The figure contains a single *spider*, which has two *feet*, and which *inhabits* the region **a**, with a foot in the zones **a∩b** and **a-b**. Spiders have a single foot in each of one or more zones.

Applied to UML, relationships between classes and states can be expressed as contours. Constraints are represented as graphs where nodes appear in appropriate set intersections.

The interpretation of Figure 1 is given by three sets **a**, **b** and **c**, which are subsets of a universal set, U, say.

The absence of a region corresponding to a set-theoretic combination, such as **a∩c**, implies that the combination must be empty: in this case the sets **a** and **c** must be disjoint.

Spiders provide lower bounds on the cardinality of sets: the spider that inhabits the region **a** implies that the region contains at least one element.

Shading is used to provide upper bounds. The shading of the zone **b∩c** implies that **b** and **c** are disjoint. The shading of the zone **a∩b** implies that it contains at most one element, that potential element being given by the spider with one foot in the zone.

There are no upper bounds on the cardinality of any unshaded zone in a spider diagram. In this particular case, it is possible for the set **b-(a∪c)** to contain any number of elements (including none).

The diagram shown in the Figure 1 is *unitary*; a general spider diagram is given by a propositional combination – using conjunction, disjunction and negation – of unitary diagrams. A full formal definition of the syntax and semantics of spider diagrams is given in [18].

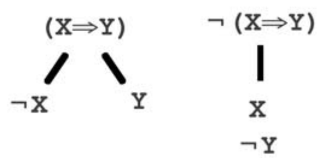
2.2 Semantic Tableaux

Semantic tableaux, [1] Section 2.6, provide an intuitive and efficient mechanism for deciding satisfiability and validity for a variety of logics. A semantic tableau for a formula is a tree, labeled at each node by a set of formulas: branches of the tree represent possible models for the formula.

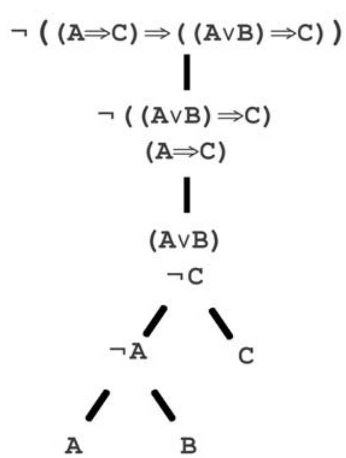
The tableau for a propositional formula is built by repeatedly applying decomposition rules to any compound formula, until only literals and their negations (or *atoms*) remain.

- A conjunction, such as A∧B, will be replaced by the pair of formulas A, B; this reflects that fact that any model of A∧B will have to make both conjuncts true.
- A disjunction like C∨D, will give rise to a split: one branch labeled C and the other D, reflecting that to satisfy a disjunction it is sufficient to satisfy one of the disjuncts.

Rules which do not cause a branch are called α–rules and branching rules are called β-rules. For each connective (e.g. implication, ⇒) there are two rules: one that decomposes the formula (X⇒Y) and the other decomposing its negation, ¬(X⇒Y). In this case, we have the rules:



Taking a larger example, we next draw the tableau for the formula ¬((A⇒C)⇒((A∨B)⇒C)). First we decompose the formula itself, giving the two formulas ¬((A∨B)⇒C) and (A⇒C). Either could be expanded, but it is usually sensible to apply α–rules before β–rules, thus delaying branching; we therefore expand ¬((A∨B)⇒C). At the next stage, two formulas remain, both with β–rules; we expand (A⇒C) and then (A∨B).

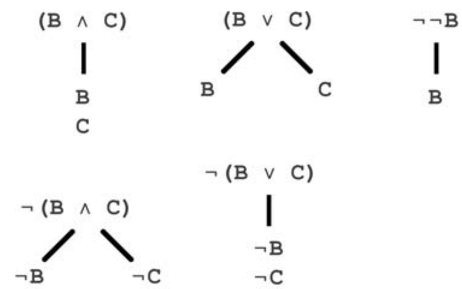


The tableau has three branches, and so embodies three potential models. Not all give models: consider the leftmost branch: that has atoms A, ¬A, C, which can't be satisfied simultaneously; similarly the rightmost branch is closed.

The central branch has atoms ¬A, B, ¬C, indicating that the formula at the root is satisfied when A and C are false and B true. Note that in building the rightmost branch it was unnecessary to expand (A∨B), since the branch was already closed,

Building this tableau has shown that the root formula is satisfiable; we can also conclude that the un-negated formula (A⇒C)⇒((A∨B)⇒C) is not valid, since its negation is satisfiable. In this way, tableaux provide a decision procedure for validity as well as for satisfiability.

For completeness we include the rules for conjunction, disjunction and negation here:



Next we look at logical equivalences between spider diagrams which will form the basis of the extension of tableaux to spider diagrams.

2.3 Reasoning Rules for spider diagrams

To build tableaux for spider diagrams we use transformation rules that allow us to transform one diagram into another logically equivalent diagram by removing, adding, or modifying diagrammatic elements. The rules are summarized below; they are based on the rules given by Shin in [16], which developed earlier work of Pierce [13].

Rule 1: Add a contour. A new contour can be drawn inside a bounding rectangle without changing the meaning of the diagram if each zone is split into two zones, inside and outside of the new contour. Each foot of a spider is replaced with a connected pair of feet, one in each new zone. Shaded zones become corresponding shaded regions.

Rule 2: Add a zone. The rule is used to add a zone absent from a diagram. The added zone is shaded to indicate that it is empty.

Rule 3: Split a spider. If a unitary diagram *d* has a spider whose habitat is formed by *n* zones, then we may replace *d* with a disjunction of *n* unitary diagrams d_1, …, d_n , each of which contains a one-footed spider inhabiting one of the zones touched by the spider *s*.

Rule 4: Expand negation. The explicit negation of a unitary diagram containing only one-footed spiders is replaced by a disjunction of (un-negated) unitary diagrams. The constraints placed on the models by shading and one footed spiders represent a conjunction of simple constraints; hence the disjunction resulting from expanding the negation.

Rules 1-4 provide the basis for diagrammatic reasoning with spider diagrams. Other rules used in building the tableau are the standard equivalences of propositional logic and compound rules built by iteratively applying combinations of rules 1-4.

Rule 5: Expand a compound diagram. This rule encapsulates the application of the tableau rules for propositional and adds the children associated by the reasoning process to 'and' and 'or' nodes. It also applies the de Morgan laws to transfer the negation on unitary diagrams. The children are computed according to propositional tableau rules [1].

Rule 6: Add contours. This rule applies Rule 1 repeatedly to add a list of contours to a collection of unitary diagrams.

Rule 7: Split spiders. Splits all the spiders used in a collection of unitary diagrams; Rule 3 is invoked several times.

Rule 8: Equalize contours. This rule is applied to a collection of unitary diagrams, which will appear in a number of different logical combinations within a tableau. Contours are added to the individual diagrams so that each diagram contains the same set of contours: the union of the initial contour sets. This rule is therefore equivalent to repeated application of Rule 1.

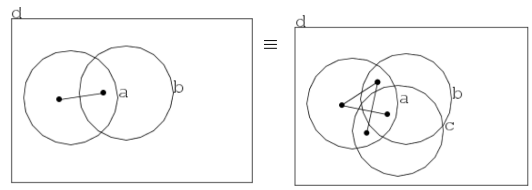
Rule 9: Equalize zones. This is the analogue of Rule 8 for zones rather than contours, and it corresponds to repeated application of Rule 2. Before adding an extra zone, contours in the diagrams need to be equalized.

Rule 10: Equalize diagrams. Invokes Rule 8 and Rule 9 to equalize both the contours and the zones.

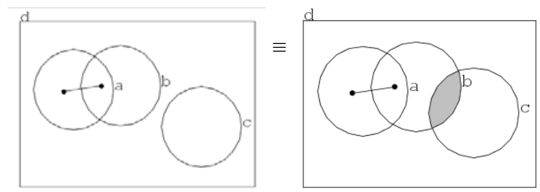
Rule 11: Expand all compound diagrams. Uses Rule 5 repeatedly until there are no more 'and' and 'or' compound diagrams.

Every unitary spider diagram is satisfiable; contradictions only occur in compound diagrams. In particular, from a unitary diagram we can read off a model by collapsing each spider to one of its feet and reading that as element of the model.

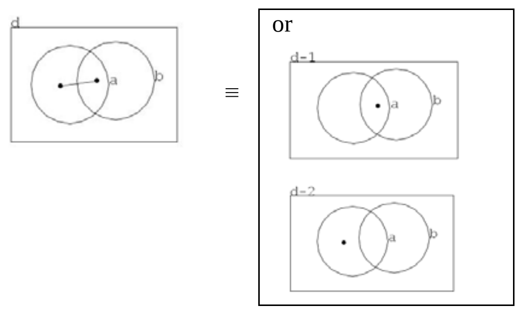
Contradictions can be explicit, as in the situation where a compound diagram contains both a diagram and its explicit negation; on the other hand, an implicit contradiction occurs in a conjunction of diagrams with conflicting constraints on a particular zone. Shading gives an upper bound on the cardinality of a zone whereas spiders provide lower bounds, and these two can conflict.



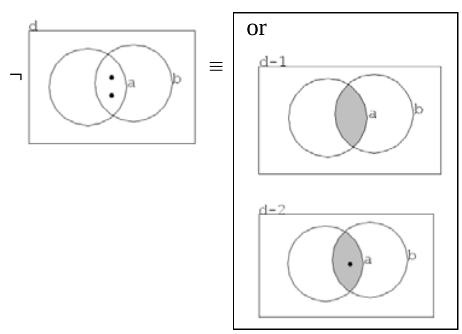
Rule 1: Adding contour *c* to diagram d



Rule 2: Adding zone b⌒c to diagram d



Rule 3: Splitting a spider



Rule 4: Negate a unitary diagram

Figure 2 Illustrating Rules 1-4

3. System Definition

This section presents a tableau system for the diagrammatic reasoning framework presented in Section 2. We begin with the definition of some terms, and then we present the rules used in the system. Then we present the algorithm that decides whether a diagram is satisfiable and analyze the satisfiability of some formulas to illustrate the algorithm. In order to design the algorithm that builds the diagrammatic tableau for spider diagrams we refer to the work presented in [1], as a framework for propositional tableaux, and [4], which presents a reasoning system for spider diagrams.

As presented in Section 2.2, a tableau is a tree, labeled with sets of formulas at each node. When spider diagrams, with diagrams as literals, replace formulas it is necessary to present the tree in a different form. We have chosen to use the JTree mechanism, which presents trees using the 'file browser' metaphor.

The tableau system is shown in the screenshots in Figure 3 and Figure 4. The upper panes of the window show the constituent unitary spider diagrams of the diagram in question; in the lower pane the tableau is shown as a JTree. Figure 3 shows a contradiction between the two diagrams d1 and d2 by outlining in bold the zone ($a \cap b$) with contradictory constraints. Figure 4 shows the effect of equalizing contours and zones between two diagrams (Rule 10 above).

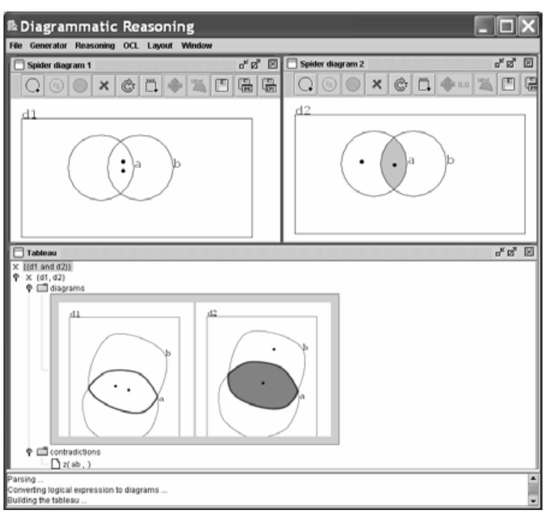


Figure 3 Equalized single-footed literals

3.1. Definitions

Definition 1. A *literal* is a unitary diagram or the negation of a unitary diagram. A unitary diagram is a positive literal and the negation of a unitary diagram is a *negative literal*. Any diagram d is the *complement* of $\neg$ d and $\neg$ d is the *complement* of d. For any diagram d, (d, $\neg$ d) is a *complementary pair of literals*.

Definition 2. A diagram that only contains spiders with one foot is a *single-footed* diagram. Otherwise it is a *non-single-footed* diagram.

Definition 3. Two diagrams d_1 and d_2 are *equalized* if they contain the same set of zones and contours. Otherwise they are *non-equalized*.

3.2. Algorithm Definition

This section presents a tableau algorithm for deciding satisfiability and hence validity for spider diagrams as presented in Section 2. This method extends semantic tableaux for the propositional calculus. We now give the construction of the semantic tableau for our diagrammatic reasoning system; the algorithm derives from the one presented in [1], Section 2.6.

Algorithm 1 (Construction of a diagrammatic tableau for spider diagrams)

Input: A diagram *d* of the spider diagrams calculus

Output: A diagrammatic tableau *T* for *d* with all the leaves marked.

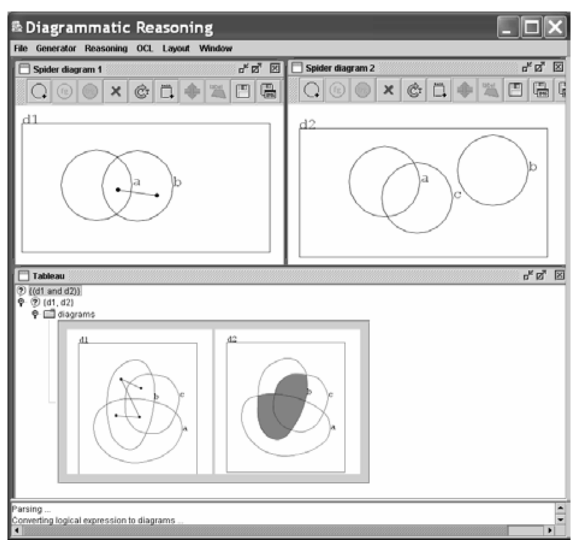


Figure 4 Equalizing rule

A diagrammatic tableau T for d is a tree for which all the nodes will be labeled with a non-empty set of diagrams. At the beginning the T consists of a single node, the root, labeled with the set $\{d\}$. The tableau is built by choosing an unmarked leaf l labeled with the set of diagrams $D(l)$ and applying one of the following rules. The construction terminates when all the leaves are marked with $\odot$ or $\times$.

- If $D(l)$ contains at least one compound diagram, *choose* a compound diagram d from $D(l)$. Iteratively create children for leaf l applying the α- and β-rules rules for propositional logic, as presented in [1].
- If $D(l)$ contains at least one negative literal, *choose* a negative literal *nd* from $D(l)$. Create children for leaf l applying the *negation* rule on *nd*.
- If $D(l)$ contains only positive literals, and they are not equalized, create children for leaf l using the *equalizing* rule presented above, by which contours are introduced into diagrams.
- If $D(l)$ contains at least one non-single-footed literal, create children for leaf l using the *splitting spiders* rule presented above, under which a diagram containing a spider is split into a disjunction of diagrams containing only single-footed spiders.
- If $D(l)$ is a set of positive single-footed literals use the contradiction rules to mark the leaf l. If there is a contradiction among the diagrams from $D(l,)$ the leaf is closed and marked with $\times$. Otherwise, it is open and marked with $\odot$.

The algorithm is not deterministic since during the expansion process of compound diagrams there is a choice of which formula to expand within the label of chosen leaf. Beside this, equalizing, negation, and spider-splitting rules generate compound diagrams, which generates non-determinism.

A diagrammatic tableau whose construction has terminated is called *completed diagrammatic tableau*. A completed diagrammatic tableau is *closed* if all the leaves are marked with $\times$. If at least one leaf is marked $\odot$, the diagrammatic tableau is *open*. Nodes below which the tableau is not completely expanded are marked ?.

The proof that the construction of a diagrammatic tableau terminates is straightforward, and is similar to the proof for semantic tableau in propositional logic [1][2]. A corollary of that result is that the order of application of the tableau rules does not affect the result of the decision procedure.

In practice, the construction of diagrammatic tableau can be made more efficient by using some of the ideas presented in [1]:

- Significant savings in space terms can be obtained if all the nodes share a diagram repository and reference elements using pointers.
- Using heuristics can make the tableau smaller. For example it is best to use α-rules before β-rules, and to split spiders only after the diagrams are equalized to avoid duplication of formulas.

Adding some derived rules will shorten the contradiction checking process. Examples include:

- If $D(l)$ is a set of literals and contains at least a *false* diagram, the leaf is closed and marked with $\times$. Otherwise other rules should be applied.
- If $D(l)$ is a set of *true* literals, the leaf is closed and marked with $\odot$. Otherwise other rules should be applied.
- If $D(l)$ contains both a diagram and its negation, then the leaf should be closed and marked with $\times$.

3.3. System Features

In this section we review the design of the tableau system, drawing attention to the various features and the motivation for their inclusion. A key aspect of a system of this sort is its usability, and to support this the system can be driven both automatically and with user intervention. We discuss system features in three broad categories now.

Logical aspects

Comprehensive set of literals and logical operators. The implemented system supports true, false, and user-defined unitary diagrams together with compound diagrams built using the propositional operators ‘not’, ‘and’, (inclusive) ‘or’, implication and equivalence. It can easily be extended to support other logical operators like exclusive or,‘xor’.

Mixed visual and textual notation. Unitary diagrams are created using a diagram editor while the compound diagrams are described using a textual notation. A parser reads the textual notation and builds an internal representation. The internal model was built using Model Driven Architecture [10] and the Kent Modeling Framework [9].

User interaction

Viewing and editing diagrams. It is possible to view and edit a population of diagrams with ease. This can be particularly important when debugging well-formedness constraints expressed using diagrammatic languages.

Familiar browsing metaphor. The system uses the J-Tree library which provides a standard interface to tree structures such as file hierarchies. This model is familiar to users from file and directory browsers.

Application of rules over diagrams. It is possible to apply transformation rules over diagrams. Selective application is supported (e.g. adding a contour, a zone, or splitting a given spider), so that one can focus on particular diagrams, without being distracted by having to check ones which are not the current focus. The feedback from rule application has been designed to be as helpful as possible.

On-the-fly application of the rules. In developing well-formedness constraints it is often very useful to be able to experiment with constraints and sub-diagrams. The system is capable of reasoning about sub-diagrams that can be then integrated into a large-scale diagram.

Backtracking. The process of diagrammatic tableau construction is non-deterministic. So, at some point one might realize that the applied rule is not appropriate. The system allows the user to go back to previous step and choose another rule.

System implementation and visualization

Diagram layout. Diagrams can be displayed using automatic layout techniques [11][14]. However, the layout process is time-consuming and so is optional. Instead, the user can view a fast embedding of a diagram, which is poorly laid out, or just view the abstract syntax as a textual collection of zones and spiders.

Visual layout. Displaying trees is always a problem because on the one hand the number of nodes tends to grow on lower levels whilst on the other the graphical space is limited to a scrollable screen. In an attempt to deal with this, and to avoid under-use of screen real estate, the system provides a mixture of vertical and horizontal display directions: tree nodes which do not contain graphic information are displayed using the vertical dimension while graphic information is displayed on the horizontal dimension.

Syntax checking for diagrams. The system detects syntactic and semantic errors prior to the construction of the tableau. The graphic editor manages the syntax errors that appear in unitary diagrams. The parser is responsible for reporting the syntax errors in the textual description of the compound diagrams. This ensures the fact the system will process only well-formed diagrams.

Link between abstract and concrete levels. After a diagram has been read using concrete syntax notations, it is transformed into an abstract representation. The reasoning is performed at the abstract syntax level. The results obtained at the abstract level are then reported to the user at the concrete level. This increases the usability and the extensibility of the reasoning system.

Diagram storage. The system offers the possibility of persistent storage for diagrams. This is a useful facility, especially in the case of large-scale systems.

3.4. Example

Figure 5 contains the tableau after the expansion of the top-level "or" and some marking. A contradiction has been detected in one branch, in the zone shown with a thick border. However, the tableau as a whole is still in an "undefined" state: more rules need to be applied in order to decide if the tableau is open or not.

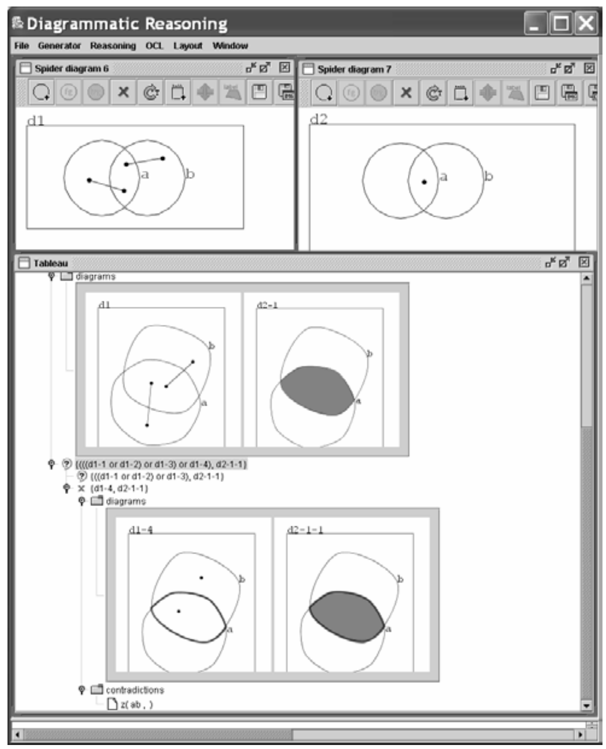


Figure 5 Indicating a contradiction between unitary diagrams

4. Conclusion and Further Work

In this paper we have described a system that supports reasoning with spider diagrams. This system allows users to construct their own spider diagrams and to explore the construction of a diagrammatic tableau. In addition, it can expand automatically all the compounds diagrams within a tableau node. The diagrammatic tableau is displayed using a mixture of vertical and horizontal display directions: tree nodes that do not contain graphic information are displayed using the vertical dimension while graphic information is displayed on the horizontal dimension.

Our plan is to extend the work in this paper to the considerably more expressive constraint diagram reasoning system [8]. Ideally, we will be able to design and implement an algorithm to construct tableaux for constraint diagrams. This is only possible for a decidable system. Restricted forms of the constraint diagram notation, which include arrows and universal spiders, yield decidable systems [17].

We also plan to use a heuristic approach to generate even shorter tableaux. The heuristic algorithm would search for an optimal operation to apply. If it fails to find a solution, it could be because more operations are required, or because there is no solution.

Currently the output from our tool appears in mixture of textual and diagrammatic notation. In order to present tableaux to users as a tree of diagrams we need to create concrete diagrams from their abstract descriptions. In [3] the authors give an algorithm for drawing a class of spider diagrams from abstract descriptions. The quality of the diagram layout has been improved using iterative methods and layout metrics [11][14]. More research is required on drawing strategies for diagrams in the context of tableaux so that the diagrams appear sufficiently similar after rule application.

Acknowledgement. This work has been funded by the UK EPSRC (Engineering and Physical Sciences Research Council) under grants GR/R63509/01 and GR/R 63516/01.

5. References

[1] Ben-Ari M. Mathematical logic for computer science. Springer-Verlag 2001.

[2] Fitting M. First-order logic and automated theorem proving, Springer-Verlag 1996.

[3] Flower, J., and Howse, J. Generating Euler diagrams, In Proceedings of Diagrams 2002, pages 61-75, Springer-Verlag, 2002.

[4] Flower J, G. Stapleton G. Automated theorem proving with spider diagrams CATS'04, Computing: The Australasian Theory Symposium , Dunedin , New Zealand, January 2004.

[5] Gil J., J. Howse, and S. Kent. Formalising Spider Diagrams, Proc. IEEE Symp on Visual Languages (VL99), IEEE Press, 130-137. 1999.

[6] Hammer E.M. Logic and Visual Information, CSLI Publications. 1995.

[7] Howse J., F. Molina, and J. Taylor. SD2: A sound and complete diagrammatic reasoning system, Proc. IEEE Symp on Visual Languages (VL2000), IEEE Press, 127-136. 2000.

[8] Kent S. Constraint Diagrams: Visualizing Invariants in OO Modelling. In Proceedings of OOPSLA97, pages 327-341. ACM Press, October 1997.

[9] Kent Modeling Framework. www.cs.kent.ac.uk/projects/kmf

[10] Model Driven Architecture www.omg.org/mda

[11] Mutton P., P. Rodgers, and J. Flower. Drawing Graphs in Euler Diagrams. Proc. Diagrams 2004, pages 66-81. Springer-Verlag LNAI.

[12] Object Constraint Language. Object Management Group http://www.omg.org document ad/03-01-07.

[13] Pierce C., Collected Papers. Vol. 4. Harvard University Press.

[14] Rodgers P., P. Mutton and J.Flower. Dynamic Euler Diagram Drawing. Proc. IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'04), pages 147-156. IEEE, September 2004.

[15] The Reasoning with Diagrams project: www.cs.kent.ac.uk/projects/rwd/

[16] Shin S-J. The Logical Status of Diagrams. CUP. 1994.

[17] Stapleton G., J. Howse, and J. Taylor. A constraint diagram reasoning system. In Proceedings of Distributed Multimedia Systems, International Conference on Visual Languages and Computing (VLC '03). pp. 263-270, Miami, USA, 2003.

[18] Stapleton G., J. Howse, J. Taylor and S Thompson. What Can Spider Diagrams Say? Proc. Diagrams 2004, pages 112-127, Springer-Verlag LNAI.

[19] Unified Modeling Language. Object Management Group http://www.omg.org.

A New Language for the Visualization of Logic and Reasoning

Gem Stapleton and Simon Thompson
University of Kent
Canterbury, UK
{g.e.stapleton,s.j.thompson}@kent.ac.uk

Andrew Fish, John Howse and John Taylor
Visual Modelling Group
University of Brighton
Brighton, UK
{andrew.fish,john.howse,john.taylor}@brighton.ac.uk

Abstract

Many visual languages based on Euler diagrams have emerged for expressing relationships between sets. The expressive power of these languages varies, but the majority can only express statements involving unary relations and, sometimes, equality. We present a new visual language called Visual First Order Logic (VFOL) that was developed from work on constraint diagrams which are designed for software specification. VFOL is likely to be useful for software specification, because it is similar to constraint diagrams, and may also fit into a Z-like framework. We show that for every First Order Predicate Logic (FOPL) formula there exists a semantically equivalent VFOL diagram. The translation we give from FOPL to VFOL is natural and, as such, VFOL could also be used to teach FOPL, for example.

1 Introduction

There is a growing interest in the use of languages based on Euler diagrams for expressing and reasoning about logical statements [1, 3, 4, 6, 7, 11, 12, 13]. The majority of these languages are monadic (meaning they can only express statements involving unary relations) and, hence, very limited in expressive power. *Constraint diagrams* [9] can make statements involving binary relations (as well as unary relations) and have been used to model object oriented software systems [8, 10]. They have been designed to complement the diagrammatic theme of the Unified Modeling Language (UML). In this paper we present a new diagrammatic language called Visual First Order Logic (VFOL) that grew out of work on constraint diagrams. VFOL is likely to be useful for software specification in the context of UML, because it is similar to constraint diagrams, and may also fit naturally into a Z-like framework. Another potential application domain is for teaching logic to computer scientists.

In figure 1 there are two constraint diagrams. The asterisks, labelled s, represent universal quantification and the nodes, labelled t, represent existential quantification. In order to disambiguate the diagrams, a *reading tree* [5] is used to indicate the order in which the quantifiers are to be interpreted. The reading trees both assert that s is read before t and d_1 expresses that A and B are disjoint, every element in A is related to precisely one element, under the relation r, which is in B. The diagrams d_1 and d_2 are examples of *unitary diagrams* which can be joined together using logical connectives such as ‘and’ and ‘or’.

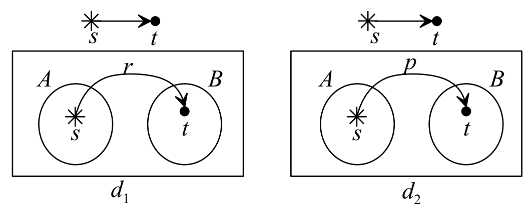


Figure 1. Two constraint diagrams.

In the constraint diagram language, it is difficult (if not impossible) to express statements such as

$$A \cap B = \emptyset \wedge \forall s \in A\, \exists t \in B \quad \{s\}.r = \{t\} \vee \{s\}.p = \{t\} \qquad (1)$$

where $\{s\}.r$ (which is called a *navigation expression*) denotes the relational image of r when the domain is restricted to $\{s\}$ (similarly for $\{s\}.p$). One reason that (1) is difficult to express is because of the disjunctive formula inside the scope of the universal quantifier. The two diagrams in figure 1 can be taken in disjunction, giving $d_1 \vee d_2$, to express

$$A \cap B = \emptyset \wedge (\forall s \in A \exists t \in B\, \{s\}.r = \{t\} \vee \forall s \in A\, \exists t \in B \{s\}.p = \{t\}),$$

but this is not semantically equivalent to (1).

Constraint diagrams are good at expressing conjunctive information inside unitary diagrams. All of the quantification occurs inside unitary diagrams, which means that First

Order Predicate Logic (FOPL) sentences involving universal quantification followed by disjunctive formulae (such as (1)) may not be realizable as constraint diagrams.

VFOL retains many features of constraint diagrams that are useful for modelling software systems. In particular, navigation expressions can still be made in VFOL. In order

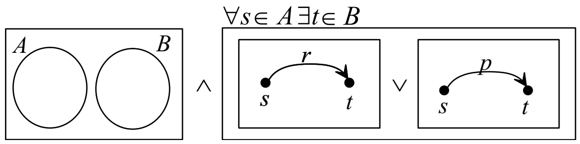


Figure 2. A VFOL diagram.

to represent relations that have arity three or greater we use multi-sourced arrows and quantification is an explicit operation which does not appear symbolically within a (unitary) diagram. Performing quantification outside diagrams also removes the need for reading trees to accompany the diagrams: the order of quantification is automatically explicit. An example of a VFOL diagram is shown in figure 2, which is semantically equivalent to statement (1) above. Unlike constraint diagrams, in VFOL distinct nodes do not necessarily denote distinct elements. This is similar to the interpretation of constant sequences in Euler/Venn diagrams: distinct constant sequences do not necessarily denote distinct individuals [13].

The syntax of VFOL and FOPL are given in section 2. The semantics of VFOL and FOPL are specified in section 3 and in section 4 we map FOPL formulae to semantically equivalent VFOL diagrams.

2. Syntax

2.1. An Alphabet

In this section, we introduce an alphabet that will be common to VFOL and FOPL. Firstly, we have a countably infinite set of **variables**, $\mathcal{V} = \{x_1, x_2, ...\}$. We define a set of **function symbols**, $\mathcal{F} = \{f_1, f_2, ...\}$, and a set of **relation symbols**, $\mathcal{R} = \{r_1, r_2, ...\}$. These two sets may be finite. A function $\alpha \colon \mathcal{F} \cup \mathcal{R} \to \mathbb{N}$ returns the arity of each symbol. Relation symbols have arity at least one. Every variable is a **term**. If f_i is a function symbol and $t_1, ..., t_{\alpha(f_i)}$ are terms then $f_i(t_1, ..., t_{\alpha(f_i)})$ is a **term**. The set of terms is denoted $\mathcal{T}$.

In this paper, we will use symbols of the form f_i and r_i in our examples. We expect users of the notation will prefer to choose sensible names for their functions and relations.

2.2. Syntax of VFOL

Relation symbols with arity one, $\mathcal{R}_1 = \{r_i \in \mathcal{R} : \alpha(r_i) = 1\}$, will be used to label contours, and we call them **given contour labels**. Function symbols with arity 0, $\mathcal{F}_0 = \{f_i \in \mathcal{F} : \alpha(f_i) = 0\}$, are constants. The remaining relation and function symbols will be used to label arrows. A special symbol, $\mathcal{U}$, represents the universal set.

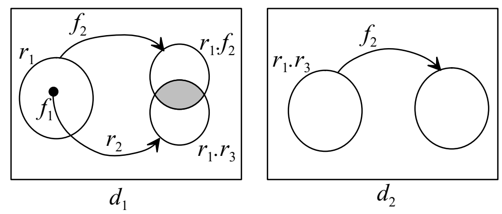


Figure 3. Two VFOL diagrams.

Example 2.1 The diagram d_1 in figure 3 contains one given contour, labelled r_1. The other two contours are *derived contours* which represent the image of a relation or function under certain restrictions. The function symbol f_1 has arity 0 and is, therefore, a constant. Locating f_1 inside r_1 expresses that $f_1 \in r_1$. The arrow sourced on r_1, labelled with the unary function symbol f_2, targets a derived contour. This arrow expresses that $r_1.f_2$ (the image of f_2 when the domain is restricted to r_1) is disjoint from r_1. This derived contour is labelled with the navigation expression $r_1.f_2$. The arrow labelled with the binary relation symbol r_2 expresses that $f_1.r_2$ equals $r_1.r_3$. By the use of shading we have expressed that $r_1.r_3$ is disjoint from $r_1.f_2$.

Derived contour labels allow us to talk about the image of a relation or function without using arrows. These labels provide an efficiency and flexibility of notation that was not present in constraint diagrams (where derived contours are never labelled). To make statements about the image in constraint diagrams, one had to first construct the image using sequences of arrows. The benefits of our new approach become apparent when constructing complex navigation expressions.

In the example above, in d_1 the derived contour label $r_1.f_2$ is redundant, since the arrow targeting $r_1.f_2$ is sourced on r_1 and labelled f_2. We will not force users of the notation to label derived contours, unless a label is essential for the interpretation of the diagram. For example, d_2 in figure 3 contains a derived contour with label $r_1.r_3$. Without this label, we could not interpret d_2 in a first order manner. The other derived contour in d_2 has not been labelled, since it represents the set $(r_1.r_3).f_2$. For space reasons, we omit the conditions under which a derived contour label is required, but they are similar to the readability criteria given for constraint diagrams in [5].

Further examples of derived contour labels are $\{x\}.f_2$ and $(\{x\} \times r_1).f_3$ where f_3 is a binary function symbol. From these simple derived contour labels we can construct more complex expressions, such as $((\{x\} \times r_1).f_3).f_2$. In order to formally define derived contour labels, we start with the set $\mathcal{DC}_0 = \{r : r \in \mathcal{R}_1 \cup \mathcal{T}\} \cup \{\mathcal{U}\}$. The elements of $\mathcal{DC}_0$ are not derived contour labels but are essential for our inductive definition below.

1. If f is a function symbol in $\mathcal{F} - \mathcal{F}_0$ and $D_1, ..., D_{\alpha(f)}$ are in $\mathcal{DC}_i$ then $((D_1 \times ... \times D_{\alpha(f)}).f)$ is in $\mathcal{DC}_{i+1}$.
2. If r is a relation symbol in $\mathcal{R} - \mathcal{R}_1$ and $D_1, ..., D_{\alpha(r)-1}$ are in $\mathcal{DC}_i$ then $((D_1 \times ... \times D_{\alpha(r)-1}).r)$ is in $\mathcal{DC}_{i+1}$.
3. Every element of $\mathcal{DC}_i$ is in $\mathcal{DC}_{i+1}$.

The set of **derived contour labels** is

$$\mathcal{DC} = \bigcup_{n \in \mathbb{N}} \mathcal{DC}_n - \mathcal{DC}_0.$$

We define $\mathcal{CL} = \mathcal{R}_1 \cup \mathcal{DC}$ to be the set of **contour labels**. The definition of $\mathcal{DC}$ could be simplified if we went against convention and defined the arity of each function symbol to be the number of inputs plus 1: we could treat relation symbols and function symbols in the same way. This is also the case for several other definitions given later in the paper.

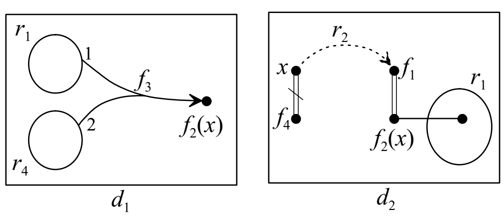


Figure 4. Multi-sourced arrows and equality.

Example 2.2 The diagram d_1 in figure 4 contains a function label f_3 that has arity 2. The arrow has two sources and the order in which these are read is indicated by labelling the arrow. The diagram expresses that $r_1 \cap r_4 = \emptyset$ and $(r_1 \times r_4).f_3 = f_2(x)$, where x is a variable which is free in d_1. Here, in our informal explanation, we have identified $f_2(x)$ with $\{f_2(x)\}$.

The arrows in a diagram can be sourced and targeted on terms, contours and the rectangle which encloses the diagram. In our formal syntax, this rectangle is denoted by $\mathcal{U}$ and represents the universal set. We define the set of sources and targets of the arrows to be $\mathcal{AST} = \mathcal{T} \cup \mathcal{CL} \cup \{\mathcal{U}\}$. Arrows are defined as follows.

1. If f is a function symbol in $\mathcal{F} - \mathcal{F}_0$, $s \in \mathcal{AST}^{\alpha(f)}$ and $t \in \mathcal{AST}$ then (f, s, t) is an **arrow**.
2. If r is a relation symbol in $\mathcal{R} - \mathcal{R}_1$, $s \in \mathcal{AST}^{\alpha(r)-1}$ and $t \in \mathcal{AST}$ then (r, s, t) is an **arrow**.

The set of all arrows is denoted $\mathcal{AR}$. The **label** of arrow (l, s, t) is l, the **source** is s and the **target** is t; the **components** of s are the elements of the set $Com(s) = \{a_i : s = (a_1, ..., a_n) \wedge 1 \leq i \leq n\}$.

We assume that the sets $\mathcal{T}$, $\mathcal{F}$, $\mathcal{R}$, $\mathcal{DC}$, $\mathcal{AR}$, and $\{\mathcal{U}\}$ are pairwise disjoint.

Example 2.3 The diagram d_2, in figure 4, expresses that $f_1 = f_2(x)$, by the use of a pair of parallel straight line segments, like an equals sign. We say that f_1 and $f_2(x)$ are *identified*. Similarly, $x \neq f_4$ and we say that x and f_4 are *separated*. The term $f_2(x)$ has a location that consists of two *zones*. In a drawn diagram, a zone can be described by a two-way partition of the contour label set. In our formalization, a zone is an ordered pair of disjoint sets of contour labels, (a, b), where a contains the zone and b excludes the zone. The diagram d_2 expresses that $f_2(x)$ is in r_1 or $\mathcal{U} - r_1$. Since $f_2(x) = f_1$, we can deduce from d_2 that $f_2(x) \in \mathcal{U} - r_1$. Finally, d_2 contains a *dashed arrow*. Dashed arrows, which are not part of the constraint diagram language, allow us to represent partial information. In the particular case here, the arrow expresses that $\{x\}.r_2$ includes f_1. In other words, x is related to (at least) f_1 under r_2.

We are now in a position to define unitary diagrams.

Definition 2.1 *A **unitary diagram** is a tuple*

$$d = \langle CL, T, SA, DA, Z, SZ, \lambda, \iota, \sigma \rangle$$

whose component parts are as follows.

1. *$CL \subseteq \mathcal{CL}$ is a finite set of contour labels.*
2. *$T \subseteq \mathcal{T}$ is a finite set of terms.*
3. *$SA \subseteq \mathcal{AR}$ is a finite set of **solid arrows** and $DA \subseteq \mathcal{AR}$ is a finite set of **dashed arrows** such that each arrow $(l, s, t) \in SA \cup DA$ satisfies $Com(s) \cup \{t\} \subseteq CL \cup T \cup \{\mathcal{U}\}$.*
4. *$Z \subseteq \{(a, b) : a \cup b = CL \wedge a \cap b = \emptyset\}$ is a set of **zones**.*
5. *$SZ \subseteq Z$ is a set of **shaded zones**.*
6. *A function $\lambda \colon T \to \mathbb{P}Z - \{\emptyset\}$ returns the **location** of each term.*
7. *A relation $\iota \subseteq T \times T$. We say that terms t_1 and t_2 are **identified** in d if $(t_1, t_2) \in \iota$ or $(t_2, t_1) \in \iota$.*
8. *A relation $\sigma \subseteq T \times T$. We say that terms t_1 and t_2 are **separated** in d if $(t_1, t_2) \in \sigma$ or $(t_2, t_1) \in \sigma$.*

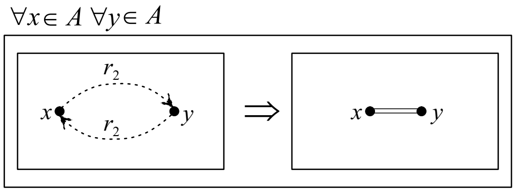


Figure 5. A compound diagram.

Unitary diagrams form the basic building blocks of *compound diagrams*.

Example 2.4 The compound diagram in figure 5 expresses that the relation r_2 is anti-symmetric when restricted to A.

To allow us to quantify over sets outside unitary diagrams, we define **set expressions**. Any contour label (given or derived) is a set expression and $\mathcal{U}$ is a set expression. If A and B are set expressions then $(A \circ B)$ is a set expression where $\circ \in \{\cup, \cap, -\}$.

Definition 2.2 *A **diagram** is defined as follows. A unitary diagram is a diagram. If d_1 and d_2 are diagrams then $\neg d_1$ and $(d_1 \circ d_2)$ where $\circ \in \{\vee, \wedge, \Rightarrow, \Leftrightarrow\}$ are diagrams. Additionally, if x_i is a variable and A is a set expression then $(\forall x_i \in A)\ d_1$ and $(\exists x_i \in A)\ d_1$ are diagrams.*

2.3. Syntax of FOPL

We briefly summarize the syntax of FOPL. The variables and terms are elements of $\mathcal{V}$ and $\mathcal{T}$ respectively. **Atomic formulae** are of two kinds. If s and t are terms then $(s = t)$ is an atomic formula. If r is a relation symbol and $t_1, ..., t_{\alpha(r)}$ are terms then $r(t_1, ..., t_{\alpha(r)})$ is an atomic formula. **Formulae** are of four kinds. Atomic formulae are formulae. If p and q are formulae then $\neg p$ and $(p \circ q)$ are a formulae where $\circ \in \{\vee, \wedge, \Rightarrow, \Leftrightarrow\}$. Additionally, if x_i is a variable then $\exists x_i\ p$ and $\forall x_i\ p$ are formulae.

3. Semantics

So far, we have given the syntax of VFOL and FOPL. We shall assume the standard semantic interpretation of FOPL formulae (see, for example, [2]). In VFOL, we briefly note that contour labels represent sets, terms represent elements (although in our formalization they represent singleton sets) and arrow labels represent relations or functions. An arrow, together with its source and target, represents a property of the relation or function represented by its label. We note that dashed arrows are syntactic sugar. This section formalizes the semantics.

Definition 3.1 *A **structure**, S, is a non-empty set U, called the **domain** of S, together with a single element subset of U for each $f \in \mathcal{F}_0$, a function $S(f)\colon U^{\alpha(f)} \to U$ for each $f \in \mathcal{F} - \mathcal{F}_0$, and a relation $S(r) \subseteq U^{\alpha(r)}$ for each $r \in \mathcal{R}$.*

Arrows give information about the image of a relation (or function) when the domain is restricted to the (set represented by) the arrows source.

Definition 3.2 *Let U denote the universal set and let f be a function. The **image** of f is the set*

$$im(f) = \{a_{\alpha(f)+1} : \exists a_1, ..., a_{\alpha(f)}\ (a_1, ..., a_{\alpha(f)+1}) \in f\}.$$

*Let A be a subset of $U^{\alpha(f)}$. We define $A.f$ to be the image of f with the domain restricted to A: $A.f = im(f|_A)$. Let r be a relation. We define the **image** of r to be*

$$im(r) = \{a_{\alpha(r)} : \exists a_1, ..., a_{\alpha(r)-1}\ (a_1, ..., a_{\alpha(r)}) \in r\}.$$

Let A be a subset of $U^{\alpha(r)-1}$. We define the image of r with the domain restricted to A to be

$$A.r = im(r \cap (A \times U)).$$

We wish to identify when a structure satisfies a VFOL diagram. In order to do so, we will interpret the component parts of the diagram, illustrated in the following example.

Example 3.1 Let S be a structure and consider the diagrams in figure 4. Some components of d_1 are interpreted in S: $S(r_1)$, $S(r_4)$, $S(f_3)$ and $S(f_2)$. We interpret $\mathcal{U}$ as the universal set: $S(\mathcal{U}) = U$. The term $f_2(x)$ is located outside both r_1 and r_4 and we associate with d_1 a *terms condition*:

$$\{x\}.S(f_2) \subseteq S(\mathcal{U}) - (S(r_1) \cup S(r_4)).$$

The solid arrow expresses

$$(S(r_1) \times S(r_4)).S(f_3) = \{x\}.S(f_2)$$

and this is called the *solid arrows condition*. The placement of r_1 and r_4 expresses that $S(r_1) \cap S(r_4) = \emptyset$. To capture this, we define the *plane tiling condition*, which asserts that the union of the sets represented by the zones is the universal set. For d_1, the plane tiling condition is:

$$(S(\mathcal{U}) - (S(r_1) \cup S(r_4))) \cup (S(r_1) \cap (S(\mathcal{U}) - S(r_4))) \cup (S(r_4) \cap (S(\mathcal{U}) - S(r_1))) = S(\mathcal{U}).$$

In d_2, the dashed arrow expresses $\{x\}.S(r_2) \supseteq S(f_1)$. The terms f_1 and $f_2(x)$ are identified, which asserts $S(f_1) = \{x\}.S(f_2)$. Separated terms denote distinct elements (strictly, singleton sets), so $\{x\} \neq S(f_4)$. Additionally, d_2 has the terms condition

$$\{x\} \subseteq S(\mathcal{U}) - S(r_1) \wedge S(f_4) \subseteq S(\mathcal{U}) - S(r_1) \wedge S(f_1) \subseteq S(\mathcal{U}) - S(r_1) \wedge \{x\}.S(f_2) \subseteq S(\mathcal{U}).$$

To facilitate the construction of a set of conditions that will allow us to determine whether a structure is a model for a diagram , we overload S. The result will include symbolic statements since, in general, the overloading contains uninterpreted variables.

Definition 3.3 *Let S be a structure with domain U. We overload S and define the following.*

1. *Universe: $S(\mathcal{U}) = U$.*
2. *Set expressions: If $A \circ B$ is a set expression where $\circ \in \{\cup, \cap, -\}$ then $S(A \circ B) = (S(A) \circ S(B))$.*
3. *Variables: for each $x_i \in \mathcal{V}$, $S(x_i) = \{x_i\}$.*
4. *Terms: for each $t \in \mathcal{T}$, if t is a constant or variable then $S(t)$ is already defined. Otherwise t is of the form $f_i(t_1, ..., t_{\alpha(f_i)})$ for some $f_i \in \mathcal{F} - \mathcal{F}_0$ and terms $t_1, ..., t_{\alpha(f_i)}$ and we define*
$$S(t) = ((S(t_1) \times ... \times S(t_{\alpha(f_i)})).S(f_i)).$$
5. *Derived contour labels: let D be a derived contour label. Then D is of the form $((D_1 \times ... \times D_n).g)$ and we define $S(D)$ recursively:*
$$S(D) = ((S(D_1) \times ... \times S(D_n)).S(g)).$$
6. *Zones: for each zone (a, b) we define*
$$S(a, b) = \bigcap_{l \in a} S(l) \cap (S(\mathcal{U}) - \bigcup_{l \in b} S(l)).$$
We define the union (intersection) over the empty set to be the empty set (the domain).
7. *Sets of zones: for each set of zones, $\mathcal{Z}$, $S(\mathcal{Z}) = \bigcup_{(a,b) \in \mathcal{Z}} S(a, b)$.*

Definition 3.4 *Let d be a diagram. The **semantics predicate**, denoted $P_d(S)$, for d is defined as follows. If d is a unitary diagram then $P_d(S)$ is the conjunction of the following conditions.*

1. ***Terms Condition*** *Terms denote elements (strictly, singleton sets) in the sets represented by their locations:*
$$\bigwedge_{t \in T} S(t) \subseteq S(\lambda(t)).$$
2. ***Solid Arrows Condition*** *Each solid arrow expresses that, when the domain is restricted to the source, the image of the label equals the target:*
$$\bigwedge_{(l,s,t) \in SA} S(s).S(l) = S(t).$$
3. ***Dashed Arrows Condition*** *Each dashed arrow expresses that, when the domain is restricted to the source, the image of the label is a superset of the target:*
$$\bigwedge_{(l,s,t) \in DA} S(s).S(l) \supseteq S(t).$$
4. ***Plane Tiling Condition*** *The union of the sets represented by the zones is the universal set:*
$$\bigcup_{z \in Z} S(z) = S(\mathcal{U}).$$
5. ***Shading Condition*** *The sets represented by shaded zones contain only elements represented by terms in the diagram:*
$$\bigwedge_{z \in SZ} S(z) \subseteq \bigcup_{t \in T} S(t).$$
6. ***Equality Condition*** *Terms that are identified represent the same elements:*
$$\bigwedge_{(t_i,t_j) \in \iota} S(t_i) = S(t_j).$$
7. ***Distinctness Condition*** *Terms that are separated represent distinct elements:*
$$\bigwedge_{(t_i,t_j) \in \sigma} S(t_i) \neq S(t_j).$$

*If $d = \neg d_1$ for some d_1 then $P_d(S) = \neg P_{d_1}(S)$. If $d = (d_1 \circ d_2)$ for some d_1, d_2 and $\circ \in \{\vee, \wedge, \Rightarrow, \Leftrightarrow\}$ then $P_d(S) = (P_{d_1}(S) \circ P_{d_2}(S))$. If $d = (Qx_i \in A)\, d_1$ for some $Q \in \{\forall, \exists\}$, variable x_i and set expression A then $P_d(S) = (Qx_i \in S(A))\, P_{d_1}(S)$. If $P_d(S)$ is true then S is a **model** for d.*

4. Mapping from FOPL to VFOL

A FOPL formula and a VFOL diagram are **semantically equivalent** when they have the same models. In order to show that FOPL is at most as expressive as VFOL we will map formulae to semantically equivalent diagrams. For example, the FOPL formula, p,

$$\exists x \exists y\, \neg(x = y) \Rightarrow (r_1(x) \wedge r_4(x, y, z))$$

is semantically equivalent to the diagram in figure 6. There is an obvious mapping from the atomic parts of p to the unitary parts of the diagram.

Definition 4.1 *Define a function, $\mathcal{E}$, which maps formula p to diagram d as follows.*

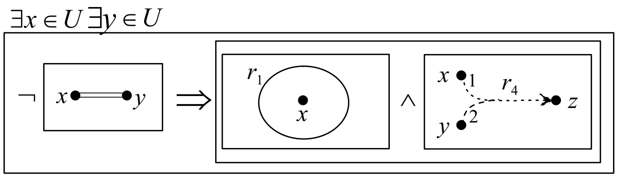


Figure 6. Mapping FOPL to VFOL.

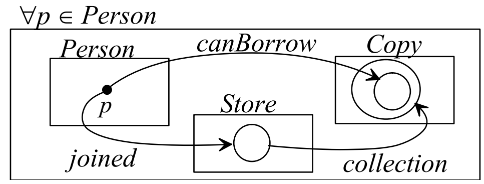


Figure 7. Specifying Software Systems.

1. p is atomic. *Then p can be one of three types and $\mathcal{E}(p) = d$ is a unitary diagram. In each case we only specify the non-empty components of d.*
   - *(a)* p is of the form $(s = t)$ for terms s and t. *The terms are $T = \{s, t\}$. The zones are $Z = \{(\emptyset, \emptyset)\}$. The locations of the terms are $\lambda(s) = Z$ and $\lambda(t) = Z$. The terms are identified, $\tau = \{(s, t)\}$.*
   - *(b)* p is of the form $r(t)$ for some $r \in \mathcal{R}_1$ and term t. *The labels are $CL = \{r\}$. The terms are $T = \{t\}$. The zones are $Z = \{(\{r\}, \emptyset), (\emptyset, \{r\})\}$. The location of the term is $\lambda(t) = \{(\{r\}, \emptyset)\}$.*
   - *(c)* p is of the form $r(t_1, ..., t_{\alpha(r)})$ for some $r \in \mathcal{R} - \mathcal{R}_1$ and terms $t_1, ..., t_{\alpha(r)}$. *The terms are $T = \{t_1, ..., t_{\alpha(r)}\}$. The zones are $Z = \{(\emptyset, \emptyset)\}$. The dashed arrows are $DA = \{(r, (t_1, ..., t_{\alpha(r)-1}), t_{\alpha(r)})\}$. The locations of the terms are, for each t_i $(1 \leq i \leq \alpha(r))$ $\lambda(t_i) = \{(\emptyset, \emptyset)\}$.*
2. p is of the form $\neg q$ where q is a formula.$\mathcal{E}(p) = \neg\mathcal{E}(q)$.
3. p is of the form $(q \circ s)$ where q and s are formulae and $\circ \in \{\vee, \wedge, \Rightarrow, \Leftrightarrow\}$. $\mathcal{E}(p) = (\mathcal{E}(q) \circ \mathcal{E}(s))$.
4. p is of the form $Qx_i\ q$ where $Q \in \{\forall, \exists\}$, x_i is a variable and q is a formula. $\mathcal{E}(p) = (Qx_i \in \mathcal{U})\ \mathcal{E}(q)$.

Theorem 4.1 *Any formula f is semantically equivalent to $\mathcal{E}(f)$.*

5. Conclusion

In this paper we have introduced VFOL which is capable of expressing any FOPL formula. We anticipate that VFOL will be useful for software specification and may also be useful for teaching logic and reasoning. An invariant we may wish to write when modelling a video rental store can be seen in figure 7. The diagram asserts that people can only borrow copies from stores they have joined.

We are developing a set of sound and complete reasoning rules for VFOL. Some of these rules will transform a diagram into a semantically equivalent diagram whose unitary parts correspond to atomic formulae. We plan to write diagrammatic versions of FOPL rules to give a complete set for VFOL and also extend to a second order language.

Acknowledgment This work is supported by the UK EPSRC grant numbers GR/R63516 and GR/R63509 for the Reasoning with Diagrams project. Thanks to Chris John for comments on earlier drafts of this paper.

References

[1] J. Barwise and E. Hammer. Diagrams and the concept of logical system. In G. Allwein and J. Barwise, editors, *Logical Reasoning with Diagrams*. OUP, 1996.

[2] S. N. Burris. *Logic for Mathematics and Computer Science*. Prentice Hall, 1998.

[3] L. Choudhury and M. K. Chakraborty. On extending Venn diagrams by augmenting names of individuals. In *Proceedings of Diagrams 2004*, pages 142–146, Cambridge, UK, March 2004. Springer-Verlag.

[4] A. Fish and J. Flower. Investigating reasoning with constraint diagrams. In *Visual Languages and Formal Methods*, volume 127 of *ENTCS*, pages 63–69, Rome, Italy, 2004. Elsevier.

[5] A. Fish, J. Flower, and J. Howse. The semantics of augmented constraint diagrams. *Journal of Visual Languages and Computing*, to appear, 2005.

[6] E. Hammer. *Logic and Visual Information*. CSLI Publications, 1995.

[7] J. Howse, F. Molina, J. Taylor, S. Kent, and J. Gil. Spider diagrams: A diagrammatic reasoning system. *Journal of Visual Languages and Computing*, 12(3):299–324, June 2001.

[8] J. Howse and S. Schuman. Precise visual modelling. *Journal of Software and Systems Modelling, to appear*, 2005.

[9] S. Kent. Constraint diagrams: Visualizing invariants in object oriented modelling. In *Proceedings of OOPSLA97*, pages 327–341. ACM Press, October 1997.

[10] S.-K. Kim and D. Carrington. Visualization of formal specifications. In *6th Aisa Pacific Software Engineering Conference*, pages 102–109. IEEE Computer Society Press, 1999.

[11] S.-J. Shin. *The Logical Status of Diagrams*. Cambridge University Press, 1994.

[12] G. Stapleton, J. Howse, and J. Taylor. A constraint diagram reasoning system. In *Proceedings of International Conference on Visual Languages and Computing*, pages 263–270. Knowledge Systems Insitute, 2003.

[13] N. Swoboda and G. Allwein. Using DAG transformations to verify Euler/Venn homogeneous and Euler/Venn FOL heterogeneous rules of inference. *Journal of Software and System Modelling*, 3(2):136–149, May 2004.

Transfer of Problem-Solving Strategy Using the Cognitive Visual Language

Jim Davies, Ashok K. Goel and Nancy J. Nersessian

Abstract— **We present Covlan, a visual language intended to model the diagrammatic memories of human beings, and Galatea, a computational model of analogical visual problem solving. Galatea and the experimental participants modeled in it show that 1) problem-solving procedures can be effectively represented with a visual language, and 2) the successful transfer of strongly-ordered procedures in which new objects are created requires the reasoner to generate intermediate knowledge states and mappings between the intermediate knowledge states of the source and target cases. We describe Galatea and Covlan, a model created with it, and related work.**

Index Terms—**Artificial Intelligence, Cognitive Science, Visual Languages**

I. Introduction

Visual Languages have been created for fields such as programming, diagrammatic reasoning, and cognitive modeling. In this work we describe Covlan, the Cognitive Visual Language, which is intended to describe visually-represented episodes of problem-solving procedures in human beings.

Some domains are inherently non-visual, but might be visually represented all the same. For example, effectively connecting a battery to some wires might be represented, among other ways, functionally (the battery needs to be physically touching the metal of the wire to conduct electricity) or visually (the image of the wire is adjacent to the image of the battery.) Even though other kinds of knowledge and representations might be used to reason about these domains, human beings appear to experience visual imagery when reasoning about them. Experimental evidence indicates that visual knowledge often plays an important role in human problem solving [7, 2, 11]. There is also documentary evidence for visual reasoning in scientific problem solving (e.g. [12]). Further, psychological evidence suggests that *analogical* problem solving is facilitated by animations [13], diagrams [1] as well as visually evocative phrases in stimuli [9]. These results suggest that visual representations have an important function in cognition, and that problem-solving procedures might be usefully represented with a visual language. Our hypothesis, then, is that problem-solving procedures can be effectively represented using a visual language.

Covlan is a model of the symbolic visual memories people have of problem-solving procedures. We have designed and implemented Galatea, a computer program that uses analogs represented in Covlan to infer problem-solving solutions to new problems [4]. To support Galatea as a model of human visual problem-solving, we modeled four experimental participants who solved a visual analogy problem, one of which we will describe here.

Dr. David Craig ran 34 participants in an analogical transfer experiment [3]. Participants were shown a problem-solving solution with a laboratory, presented with text and a diagram. They were asked to solve an analogous problem with a weed-trimmer, presented with text only. Of these, 17 participants (in three conditions) correctly described the analogous solution. All participants were asked to draw a diagram to illustrate their suggested solutions (See Figure 1 for the participant stimuli). A laboratory clean room strategy is transferred by adding redundant doors to a weed-trimmer arm so that it can pass through street signs. The analogous solution is to design an arm with two latching doors, so that while one is open to let the sign pass, the other stays closed to support the arm and trimmer (See Figure 2 for one participant's data). Participants produced diagrams describing their solutions to the problems. We modeled four of these experimental participants in Galatea: L14, L15, L16, and L22. We will describe in detail our model of one of these participants, L14, in this paper, and briefly describe the results of modeling the other three. We use our ability to model these participant data as an evaluation of Covlan and Galatea as a cognitive theory.

II. Galatea

A. Overview

Analogical problem solving involves several steps: An intelligent agent (e.g., a human or an artificial intelligence program) starts with an unsolved problem, the *target*, and *retrieves* a similar *source*, or *base*, episode of a solved

Manuscript received March 31, 2005.

J. Davies is a postdoctoral fellow at the School of Computing, Queen's University, Kingston, Ontario, Canada, K7L 3N6. (1-613-532-8828; fax: 1-613-533-6513; e-mail: jim@jimdavies.org).

A. K. Goel is an associate professor in the College of Computing, Georgia Institute of Technology, Atlanta, Georgia, USA. (404-894-4994; e-mail: goel@cc.gatech.edu).

N. J. Nersessian is a professor in the College of Computing, Georgia Institute of Technology, Atlanta, Georgia, USA. (404-894-1232; e-mail: nancyn@cc.gatech.edu).

problem. Then the elements of the source are *mapped* to the elements of the target. This means finding alignments between the sub-parts of the problems. Then the source's solution is *transferred* to the target, perhaps with some adaptation. Then the solution is *evaluated* and finally *stored* in memory. Most analogical AI systems model analogical mapping. Galatea models the transfer stage of analogical problem solving.

The modeling architecture used to model L14, one of the experimental participants, is an implemented computer program called Galatea. The issue is how an analogical problem solver might represent its diagrammatic knowledge of the source case and target problem, and how might it transfer the relevant problem-solving steps from the source to the target?

Galatea represents a source case as a series of knowledge states starting from the initial knowledge state and ending in the final or goal knowledge state. A knowledge state is represented diagrammatically in the form of shapes, their locations, sizes, and motions (if any), and the spatial relationships among the shapes.

Succeeding states in the series of knowledge states are related through visual transformations such as move, rotate, scale and decompose. Each transformation relates two knowledge states. Transfer works by applying, step by step, each transformation in the source case to the knowledge states of the target case.

B. *Knowledge Representation: Covlan*

Galatea describes visual cases using Covlan, which consists of knowledge states, primitive elements, primitive relations, primitive transformations, general visual concepts, and correspondence and transform representations. In Covlan, all knowledge is represented as propositions relating two elements with a relation.

Knowledge States: Knowledge states in Covlan are symbolic images, or *s-images*, which contain visual elements, general visual concepts, and relations between them. Knowledge states are represented by a series of *s-images*, connected with transformations.

Visual Transformations. An *s-image* in the sequence is connected to other *s-images* before and after it with visual transformations. Transformations, like ordinary functions, take arguments to specify their behavior.

These transformations control normal graphics transformations such as translation (*move-to-location*), and rotation (*rotate*). In addition there are transformations for adding and removing elements from the *s-image* (*add-element, remove-element*). Certain transformations (*start-rotating, stop-rotating, start-translation, stop-translation*) are changes to the dynamic behavior of the system under simulation. For example, *rotate* changes the initial orientation of an element, but in contrast *start-rotating* sets an element in motion.

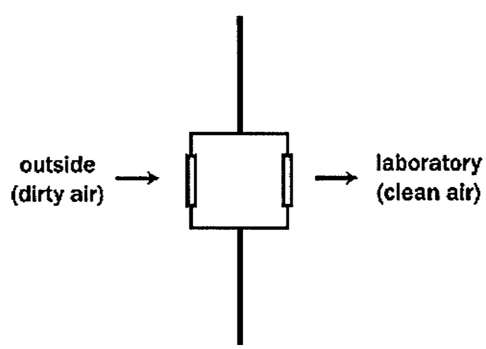
Primitive Elements are the visual objects in a diagram. The element types are *rectangle, circle, arrow, line,* and *curve.* Each element is represented as a frame with attribute slots, such as *location, size, orientation,* or *thickness.* A particular example of an element is referred to as an *element instance.*

General Visual Concepts. These act as slot values for the primitive elements as well as arguments for the visual transformations. The concepts are *location, size, thickness, speed, direction, length, distance, angle,* and *direction.* Each concept has several values it can take. For example, the *size*

Please read the two problems below. At the bottom of the page, please try to solve **Problem 2.** *Draw a diagram to show what you're thinking. The solution to* **Problem 1** *may be helpful in solving* **Problem 2.**

Problem 1: A computer chip manufacturer has designed a special lab for manufacturing microscopic devices. They have taken great care to seal off the lab from the surrounding environment in order to keep the air inside the lab free of dust and undesirable gases. The problem, though, is that whenever lab workers enter or leave the room, the seal is broken and contaminated air is allowed in. The company is trying to design a door that will allow workers to enter and leave the lab easily, while minimizing the amount of contaminated air that is let in.

Solution: Have workers enter a vestibule space before entering the lab.



Problem 2: In order to trim the weeds that grow along the side of the road, the Department of Transportation has designed a weed trimmer that attaches to the end of a long pole sticking off the side of a truck. As the truck drives down the highway, the trimmer is extended about 6 feet to the right, perfectly positioned to trim the weeds at the side of the road. The problem is that the 6-foot pole is obstructed by sign posts that are positioned at the curb in certain parts of the city. The weed-trimmer pole, in fact, is exactly 2 feet too long to clear the sign posts. Although the weed-trimmer pole could be retracted or lifted out the way to clear the sign posts, this would interfere with the weed trimming. And although the pole could bend over the top of the sign posts, this would be impractical since in some areas the signs are 15 feet tall. The Department of Transportation is trying to design a pole that can pass *through* the sign posts without stopping or changing the position of the trimmer.

Figure 1: The stimuli shown to L14 in the experiment.

In the space below, try to design a weed-trimmer pole that can pass through sign posts. Draw a diagram to illustrate what you're thinking.

The problem is 2 ft too long, so from 3½ ft to 4½ ft. Have a sliding bar that unlatches (twist) and retracts (Pull), for upcoming signs. And slides back up and latches when sign passes in. Then other arm does same and the sign passes through

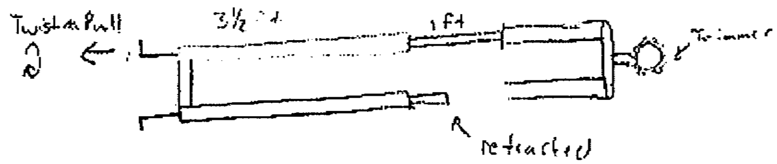


Figure 2: What L15 drew on the experiment sheet.

can be *small, medium,* or *large,* and *thickness* can be *thin, thick* or *very-thick. Location* specifies an absolute qualitative location in an *s-image* (*bottom, top, center,* etc.)

Primitive Visual Relations. This class of symbols describes how certain visual elements relate to each other and to the values taken by general visual concepts. The visual relations are *touching, above-below,* and *right-of-left-of.* The motion relation is *rotation.*

Correspondence and Transform Representations. The knowledge of which objects in one *s-image* correspond to which objects in another is a *mapping,* which consists of a set of alignments between objects. Different sets of alignments compose different mappings. The i^{th} *s-image* in the source and the i^{th} *s-image* in the target have a *correspondence* between them; each correspondence can have any number of *mappings* associated with it (determining which mapping is the best is the "mapping problem.") The correspondence and mapping between the initial *s-images* (i=1) in the source and target is given as part of the input to Galatea; the system generates the subsequent correspondences and mappings for the subsequent *s-images.*

Similarly, successive *s-images* in a series have *transform-connections.* These are needed so that Galatea can track how visual elements in a previous knowledge state change in the next.

C. Algorithm

Following is the control structure for Galatea's transfer of problem-solving procedures from a source case to the target problem. Figure 3 shows the *s-image* structure for L14's problem and solution. The algorithm below references Figure 3.

The solution procedure (for the source, and then for the target) is that the doorway mechanism gets replicated, and then moved to the correct positions. Two walls are created to complete the vestibule, and finally they are placed in the correct position so that the vestibule is complete.

1. **Identify the first *s-images* of the target and source cases.** These are the current source and target *s-images.*
2. **Identify the transformations and associated arguments in the current *s-image* of the source case.** This step finds out how the source case gets from the current *s-image* to the next *s-image.* The model of L14 involves five transformations (see Figure 3). The first transformation is *replicate.* The second transformation is *add-connections* which places the door sets in the correct position in relation to the top and bottom walls. The third and fourth transformations are *add-component,* which adds the top and bottom containment walls. The fifth transformation, another *add-connections,* places these containment walls in the correct positions in relation to the door sets and the top and bottom walls.
3. **Identify the objects of the transformations.** The object of the transformation is what *object* the transformation acts upon. For L14's first transformation, this object is the parts of the door in the first *s-image* (we'll call it *door-set-l14s1*).
4. **Identify the corresponding objects in the target problem.** In the target, the trimmer arm's door mechanism is the corresponding object.
5. **Apply the transformation with the arguments to the target problem component.** A new *s-image* is generated for the target problem to record the effects of the transformation. *Replicate* takes two arguments: some *object* and some *number-of-resultants.* In this case the *object* is *door-set-b1s1* (b1s1 means "base one, *s-image* two") and the *number-of-arguments* is two. The *replicate* is applied to the first L14 *s-image,* with the appropriate adaptation to the arguments: The mapping between the first source and target *s-images* indicates that the *door-set-b1s1* maps to the *door-set-l14s1,* so the former is used for the target's object argument. The

number *two* is a literal, so it is transferred directly. Galatea generates *door-set1-l14s2* and *door-set2-l14s2* in the next *s-image*.

The second transformation is *add-connections*. The effect of this transformation is to place the replicated door-sets in the correct spatial relationships with the other element instances. It takes *connection-sets-set-b1s3* as the *connection/connection-set* argument. This is a set containing four connections. Galatea uses a function to recursively retrieve all connection and set proposition members of this set. These propositions are put through a function which creates new propositions for the target. Each proposition's relation and literals are kept the same. The element instance names are changed to newly-generated analogous names. For example, *door1-endpoint-b1s3* turns into *door1-endpoint-l14s3*.

Then, similarly to the replicate function, horizontal target maps are generated, and the other propositions from the previous *s-image* are instantiated in the new *s-image*.

The inputs to this transformation are *nothing* (a literal denoting that there is not any thing in the previous *s-image* that is being modified), the connection set *connection-sets-set-b1s3*, the source *s-image lab-base1-simage2*, the current and next target *s-images l14-simage2* and *l14-simage3*, the mapping *l14-simage2—l14-simage3-mapping1,* and the rest of the memory.

6. **Map the original objects to the new objects in the target case**. A transform-connection and mapping are created between the target problem *s-image* and the new *s-image* (not shown). Maps are created between the corresponding objects. In this example it would mean a map between door-sets, as well as their component objects. Galatea does not solve the mapping problem, but a mapping from the correspondences of the first *s-image* enables Galatea to automatically generate the mappings for the subsequent *s-images*.

7. **Map the new objects of the target case to the corresponding objects in the source case**. Here the parts of the door set in the target *s-image* are mapped to the parts in the second source *s-image*. This step is necessary for the later iterations (i.e. going on to another transformation and *s-image*). Otherwise the reasoner would have no way of knowing which parts of the target *s-image* the later transformations would operate on.

8. **Check to see if goal conditions are satisfied**. If they are, exit, and the solution is transferred. If not, and there are further *s-images* in the source case, set the current *s-image* equal to the next *s-image* and go to step 1.

We can now evaluate what made L14's data (Figure 2) differ from the stimulus drawing (Figure 1): L14 features a longer vestibule in the drawing than the vestibule pictured in the stimulus. In fact, there is no trimmer arm (analogous to the wall in the lab problem) in the drawing at all that is distinct from the vestibule, save a very small section, apparently to keep the spinning trimmer blade from hitting the vestibule. The entire drawing is rotated ninety degrees from the source. The single lines in the source are changed to double lines in the target. The doors also slide in and out of the vestibule walls. What's interesting about this modification is that it does not appear that this kind of door opening is possible with the diagram given of the lab in the source: Since the door is a rectangle that is thicker than the lines representing the walls, the door could not fit into the walls. In contrast L14 explicitly makes the doors and walls thick (with two lines) and makes the doors somewhat thinner. L14 adds objects to the target not found in the source: a blade and a twisting mechanism to describe how the doors can work. L14 also included numerical parameters to describe the design of the trimmer: to describe length. Finally, L14 includes some mechanistic description of how the trimmer would work.

Of these seven differences, our model successfully re-creates four of them. The *rotation* of the source is modeled by a rotation in the target start *s-image*. In the *s-image*, all spatial relationships are defined only relative to other element instances in the *s-image*. Each instance is a part of a single set which has an orientation and direction. In the case of *s-image* 1 of the target, it is facing right. Since all locations are relative, there is no problem with transfer and each *s-image* in the model of L14 is rotated to the right. The *line to double line* difference is accounted for by representing the vestibule walls with rectangles rather than with lines, as it is in the

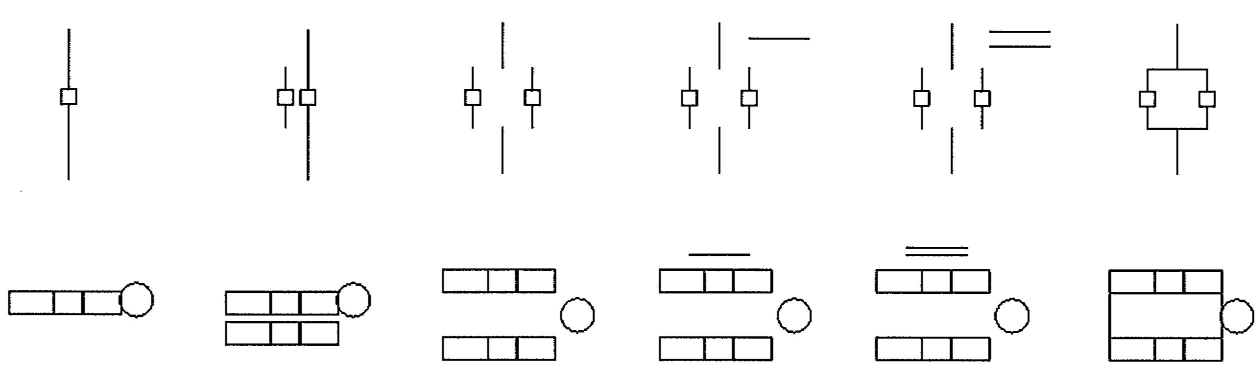


Figure 3: The Galatea Model of L14

source. Because the mapping between the source and target correctly maps the *side1* of the rectangle to the *startpoint* of its analogous line, the rectangle/line difference does not adversely affect processing transfer. The *long vestibule* difference is accounted for by specifying that the heights of the vestibule wall rectangles are *long*. In the source the vestibule wall lines are of length *medium*, but this does not interfere with transfer. The trimmer head *added object* is accounted for by adding a circle to the first *s-image* in the target.

Unaccounted for are the two bent lines emerging from the vestibule on the left side, the numeric dimensions and words describing the mechanism. Also, L14 shows one of the doors retracting, and the model does not. The model also fails to capture the double line used to connect the door sections, because the single line is transferred without adaptation from the source. This could be fixed, perhaps, by representing the argument to the *add-component* as a function referring to whatever element is used to represent another wall, rather than as a *line*.

Though we do not have the space to report the details here, we implemented three other models from the Craig et al. experiment. The four in total best represented the difficulty and variety of the participants we could have modeled. In all cases, our models accounted for the majority of the differences between the participant drawings and the stimulus given, supporting Covlan and Galatea as models of human thought.

III. Related Work

Though much work on visual languages involves visual programming aids, there are some visual languages meant to model the internal visio-spatial representations in people's minds. Liu's PI system [10] represents objects and operations in the Euclidean geometry domain, but is not intended as a cognitive model of human visual thought. GeoRep [8] uses a set of "primitive shapes" that resemble Covlan's visual elements. It is intended to be a model of human visual reasoning. Erwig and Schneider [6] represents changes for visual objects, but not in terms of actions taken on a visual system. Their system allows queries with respect to what things have happened (e.g. has a tornado ever passed through Iowa?). Covlan, to our knowledge, is the first to explicitly represent sequences of visual operations. Covlan also manages the complexity involved with, for example, adding objects and keeping track of what is done with them. And with regard to the systems that use visual languages, Galatea is the first to transfer visually represented problem-solving procedures.

IV. Conclusions

Our hypothesis, as stated in the introduction is that problem-solving procedures can be effectively represented using a visual language. We have presented the Covlan, a visual language designed to describe problem-solving situations and Galatea, the artificial intelligence system that uses Covlan representations to transfer sequences of visual actions to a target problem. Our hypothesis was supported by the evidence described above, and we had an unexpected discovery for a total of two claims.

First, problem-solving procedures can be effectively represented using a visual language. There are seven models written in Covlan and Galatea that support this claim. We described the model of L14 in this paper. In addition we modeled three additional participants from the Craig *et al.* experiment, a historical example from the scientific thinking of Maxwell [5], the fortress/tumor problem [4] and the cake/pizza problem [4]. Each of these models uses analogical reasoning to solve a problem using only visual knowledge. The fact that four of these models are based on human experimental participant data lend support to the idea that the claim might apply to human problem solving, as well as artificial analogy systems, although more empirical research would be needed to substantiate this. As shown above, most of the differences between source and target, as displayed in the participant data, were accounted for in our models.

In the course of building the models of Galatea, we discovered that the successful transfer of strongly-ordered procedures in which new objects are created requires the reasoner to generate intermediate knowledge states and mappings between the intermediate knowledge states of the source and target cases. Galatea shows why, in detail, this is so. Components of the problem are *created* by the operations, and these components are acted on by later operations. For L14's problem, for example, the door set must be replicated before the two sets can be moved in relation to one another. When the reasoner transfers the second operation of moving the door sets, how does it know what the corresponding objects are in the target? It must have some mapping to make this inference. And since one of the door sets did not exist in the start states of the problems, this mapping cannot be given as input with the initial mapping. The new knowledge state with the duplicated door set must be generated, and then a mapping must be made on the fly between it and the second knowledge state of the source case.

In conclusion, Covlan appears to be a good start for a full cognitive language of visual representations of procedural strategy.

Acknowledgment

Thank you to Dr. David Craig for the use of his experimental data.

References

[1] Beveridge, M. & Parkins, E. Visual representation in analogical problem solving. *Memory & Cognition*. 15(3) (1987) 230–237

[2] Casakin, H., Goldschmidt, G.: Expertise and the use of visual analogy: Implications for design education. *Design Studies*. (1999)

[3] Craig, D. L., Catrambone, R., Nersessian, N. J.: Perceptual simulation in analogical problem solving. In *Model-Based Reasoning: Science, Technology, & Values*. New York: Kluwer Academic / Plenum Pubishers. (2002) 167–191

[4] Davies, J., Goel, A. K.: Visual case-based reasoning I: Transfer and adaptation. *Proceedings of the 1st Indian International Conference on Artificial Intelligence*. Springer. (2003)

[5] Davies, J., Nersessian, N. J., Goel, A. K.: Visual models in analogical problem solving. *Foundations of Science, Special Issue on Model-Based Reasoning: Visual, Analogical, Simulative*. By Magnani, L. and Nersessian, N.J. (Eds.) (in press)

[6] Erwig, M., Schneider, M. *Journal of Visual Languages and Computing, Vol. 14*, No. 2, 181-211 (2003)

[7] Farah, M. J.: The neuropsychology of mental imagery: Converging evidence from brain-damaged and normal subjects. In *Spatial Cognition-Brain Bases and Development*. Erlbaum (1988)

[8] Ferguson, R. W., & Forbus, K. D. GeoRep: A flexible tool for spatial representation of line drawings, *Proceedings of the 18th National Conference on Artificial Intelligence*. Austin, Texas: AAAI Press. (2000)

[9] Gick, M. L. & Holyoak, K. J. Schema induction and analogical transfer. *Cognitive Psychology*. 12. (1980) 306–355

[10] Liu, Zhiqing, Semantics Design of a Visual Language for Constructing and Animating Geometric Objects. In the 1999 IEEE Symposium on Visual Languages, Tokyo, Japan, September (1999)

[11] Monaghan, J. M., Clement, J.: Use of computer simulation to develop mental simulations for understanding relative motion concepts. *International Journal of Science Education.21(9)*. (1999) 921–944

[12] Nersessian, N. J. Maxwell's `Newtonian aether-field'. In Nersessian, N. J. *Faraday to Einstein: Constructing Meaning in Scientific Theories*. Kluwer, Dordrecht. (1984). 68-93.

[13] Pedone, R., Hummel, J. E. & Holyoak, K. K. The use of diagrams in analogical problem solving. *Memory & Cognition*. (2001)

Global and Vector Operations in a Rule-Based Visual Language

Joseph J. Pfeiffer, Jr.
Department of Computer Science
New Mexico State University
Las Cruces, NM 88003
pfeiffer@cs.nmsu.edu

Abstract

In order to perform global operations such as robot path planning, it is necessary to support operations which are performed at any matching location in a model rather than merely at locations defined by the current location of the robot. This paper describes extensions to the rule-based geometric reasoning language Isaac to permit global operations, without violating the rule-based nature of the language. An example of the use of the extended language to define a path-planning algorithm is presented.

1. Introduction

The Isaac rule-based visual language has been developed to define reactive robot behaviors using a purely geometric notation, and to allow environment mapping. A weakness in the language formulation has been that more sophisticated algorithms, such as path planning operations, could not be defined within the constraints of the language.

In this work, we consider the addition of two features to Isaac to address this deficiency. First, we consider rules which are not robot-centric; that is, rules which can be applied anywhere in the world model rather than just at locations relative to the robot. Second, we enhance the world model itself by adding vector fields, permitting the robot to follow a path defined in the model.

Following this Introduction, we describe the Isaac visual programming language in Section 2. Section 3 describes the enhancements being made to Isaac. A detailed example, showing the use of global rules and vector planes, is given in Section 4. Finally, Section 5 presents some preliminary conclusions.

2. Isaac

Isaac is a rule-based visual language for geometric reasoning in uncertain environments, specifically intended for use with mobile robots. Detailed descriptions of the language, its processing model, and the techniques used in visualizing world models used in Isaac can be found in [4] and [2]. A description of a prototype execution engine for the language is described in [3].

Briefly, Isaac uses rules defined solely in geometric terms for robot control. A typical rule, which would be used for obstacle avoidance, is shown in Figure 1.

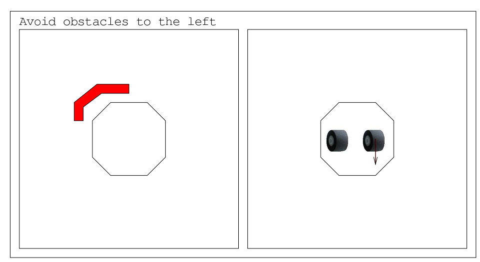


Figure 1. Obstacle Avoidance Rule

The rule's preconditions are shown in the left pane of the rule, and its postconditions in the right. In both panes, the robot is represented as an octagon; the robot's orientation is such that "forward" is represented as "up" in the rule. The left (precondition) pane shows a red[1] region forward and to the left of the robot; the right pane shows the result of applying this rule, which is to stop the robot's left motor and set its right motor to run in reverse.

These rules are used in combination with a world model defined in terms similar to Moravec's evidence (or occupancy) grids[1]. In an Isaac world model, each point in the model represents a vector of object types which may or may not be present at that point. Typical object types might be obstacles (either known *a priori* or discovered through sensor returns) or "virtual" objects such as a trail marking

[1]The colors are described as in the on-line version of this paper. Although the paper will be reproduced in greyscale, figures will still be described using the colors present in the on-line version.

where the robot has been. Each of the object types at a point is maintained independently using Dempster-Shafer[5] belief values, and a frame of discernment $\{T, F\}$ in which T is the extent to which it is believed the object is present at the point, and F is the extent to which it is believed the object is not present.

The rule in Figure 1 would be appropriate to a world model in which obstacles detected by the robot's sensors appear as red objects in the world model (the input/output system used by Isaac, and its integration with the rule execution engine, is described in [4]). The interpretation of the rule is that, if the red area in the left pane of the rule intersects a red region (*i.e.* an obstacle) in the world model, then the rule is applied.

Rules can be defined which will directly control the robot (as above), or to modify the world model itself. Rules in this latter category will be shown later in this paper.

3. Global Rules and Vector Planes

3.1. Global Rules

A global rule is specified in Isaac by not including a robot outline in the rule's specification. The global operation can then be performed at any matching location in the world model. In robo-centric rules, the presence of the robot outline imposes an orientation on the rule: the rule is specified relative to the orientation of the robot, and is rotated with the robot when applied. Global operations do not have the presence of the robot to impose an orientation, so they are being treated as isotropic – that is, the specified rule matches as defined by the rule, and also matches if rotated a multiple of 90° from its definition.

An example of a global Isaac operation is shown in Figure 2. In this operation, it is assumed that obstacles are coded red (the obstacle location could either be *a priori* information from an input map, or new information gained as a result of sensor inputs to the robot). The operation performs a dilation, inserting an orange square of the specified size matching every red point in the original model.

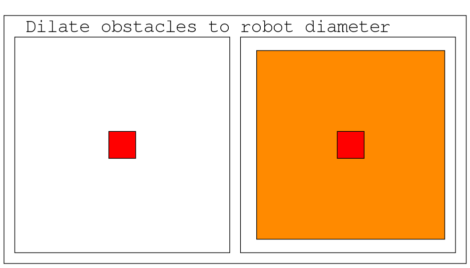


Figure 2. Dilation Operation

An example of applying this rule to a map will be shown later, in combination with rules performing path-planning.

3.2. Vector Planes

Polygons in Isaac represent the presence of features in the world model, with Dempster-Shafer values for the belief in the presence and the belief in the absence of the feature at the location. A vector plane also has a unit vector at each point in the plane, specifying a direction. When a rule is applied which modifies a vector plane, the direction is updated according to the weighted average of the beliefs in the presence of the point. An example of three rules including vectors is shown in Figure 3. In these rules, the blue square in the left hand side denotes the path, and the short black line denotes the direction of the path. The empty orange square denotes the requirement that no dilated obstacle exist in that region of the plane. On the right hand side, the path has been extended.

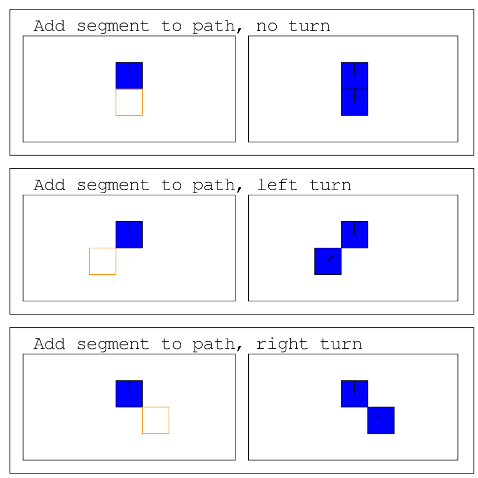


Figure 3. Path Planning Rule to Extend Path Backwards

All three of these rules correspond to a situation in which the existing path is in one of the four primary compass directions. In the first rule, the path can be extended in the same direction. In the second and third rules, diagonal turns to the left and right are also possible. Three similar rules would handle the similar case of extending diagonal path elements. As will be seen in Section 4, this rule is part of a backward-chaining path planning algorithm: the path is extended backward from the goal point to the initial robot position.

Examples of rules causing a robot to travel as specified by a vector plane are shown in Figure 4. When a rule whose precondition includes an orientation, the degree to which it matches is weighted by the cosine of the angle between the orientation of the rule and the direction in the vector field if the angle is between 0 and $\pm 90^o$, and does not match at all if the angle is obtuse. In the three rules shown, the first rule will completely match if the robot is heading in the same direction as the path, the second will completely match if the robot is travelling 90^o to the right of the path, and the third will completely match if the robot is travelling $90^o$to the left of the path. The weighted average of rules will be combined, resulting in the robot correcting its direction of travel to correspond to the path.

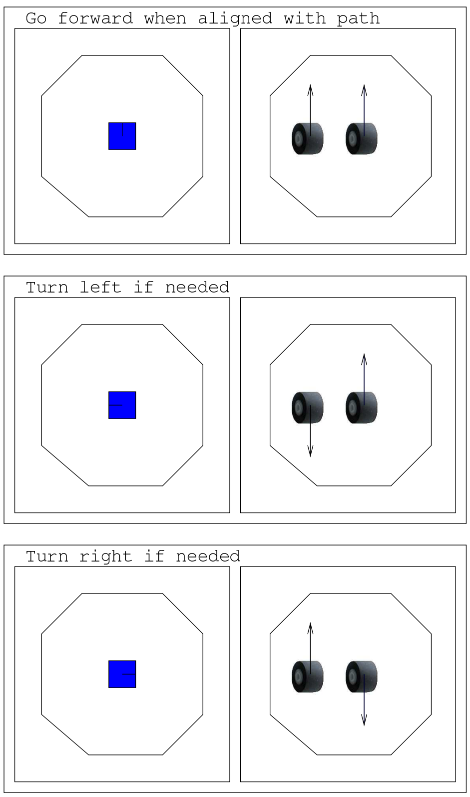


Figure 4. Rules for Following Path

The visualization of vector fields is likewise an extension of the model visualization already present in Isaac. The use of hue, brightness, and saturation to visualize belief and disbelief in obstacles, is described in [2].

To visualize vector fields, the approach is augmented by drawing lines as specified by the direction of the visualization. The model display is divided into small squares. If the center point of a square has a vector direction, a short line is drawn from the center in that direction.

4. Detailed Example: Path Planning

This section presents a detailed example of the use of global operations and vector fields to implement a stream function path planning algorithm[6]. The ruleset for the path planning problem consists of the dilation rule in Figure 2, the three path-planning rules shown in Figure 3 (and the three analogous diagonal rules mentioned in Section 3.2), and the path following rules from Figure 4.

The original model is as shown in Figure 5.

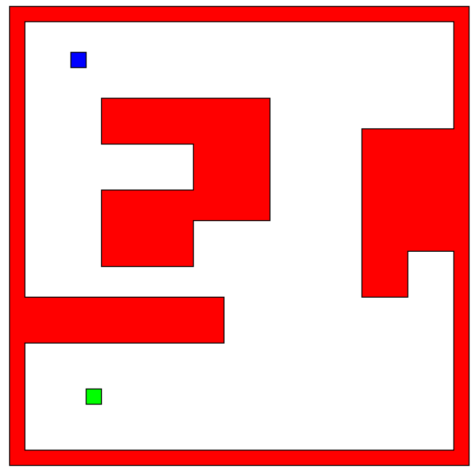


Figure 5. Original World Model

On the first rule execution cycle, the dilation rule dilates the obstacles, and the path planning rules create the last step in the path: the step that actually achieves the goal. The result of this execution cycle is shown in Figure 6.

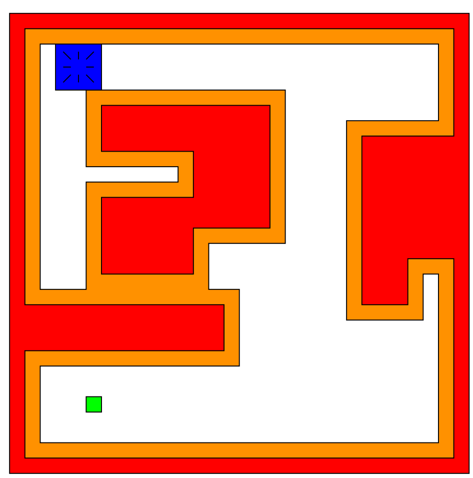


Figure 6. First Step in Path Planning

As the dilated obstacle avoidance region is not the same color as the original obstacle region, it does not continue to dilate. However, the path-planning region continues to expand. Following ten iterations it appears as shown in Figure 7.

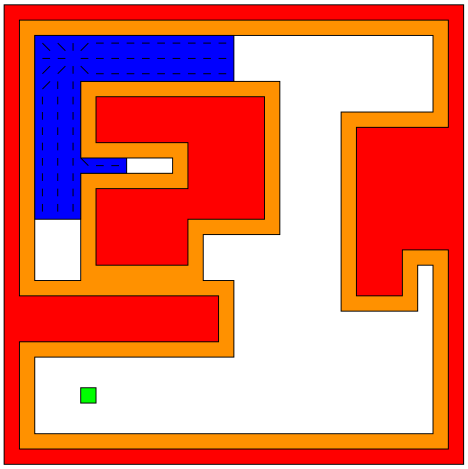


Figure 7. Path Planning After Ten Iterations

Once the planned path reaches the robot, the path following operations shown in Figure 4 are enabled, and the robot proceeds to the target. The final vector field, and the path taken by the robot, are shown in Figure 8.

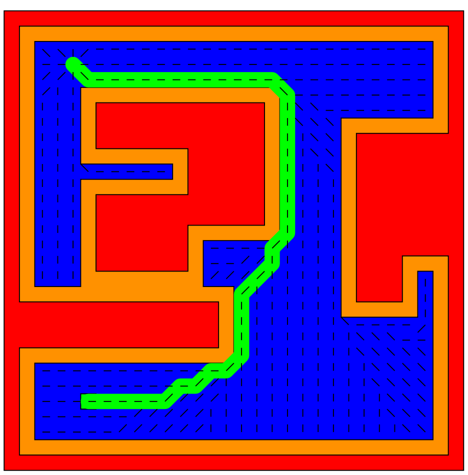


Figure 8. Path Followed by Robot

This algorithm is "self healing" in the event of the discovery of additional obstacles in the course of the robot's travel to the goal point. If a red obstacle point is added to the model, the dilation rule is enabled, creating an orange area which is ineligible for path planning. The path will then "re-propagate" around the new orange area.

5. Conclusions and Future Work

This paper has described enhancements currently being added to Isaac visual programming language to permit a more powerful range of algorithms than those of the language's initial description. It is expected that this will increase the utility of the language in both research and instructional environments.

The enhancements to Isaac described in this paper are currently being implemented. At the same time, we are reconsidering the rule execution engine itself with an eye to reducing the amount of computation needed on each execution cycle, focusing on geometric analogies to the well-known RETE and TREAT algorithms.

References

[1] Hans Moravec and A. E. Elfes. High resolution maps from wide angle sonar. In *Proceedings of the 1985 IEEE International Conference on Robotics and Automation"*, pages 116–121, March 1985.

[2] Joseph J. Pfeiffer, Jr. Using brightness and saturation to visualize belief and uncertainty. In Mary Hegarty, Bernd Meyer, and N. Hari Narayanan, editors, *Diagrammatic Representation and Inference, Second International Conference, Diagrams 2002*, pages 279–289, April 2002.

[3] Joseph J. Pfeiffer, Jr. A prototype inference engine for rule-based geometric reasoning. In Alan Blackwell, Kim Marriott, and Atsushi Shimojima, editors, *Diagrammatic Representation and Inference: Third International Conference, Diagrams 2004*, pages 216–226, Cambridge, UK, March 2004. Springer-Verlag.

[4] Joseph J. Pfeiffer, Jr., Rick L. Vinyard, Jr., and Bernardo Margolis. A common framework for input, processing, and output in a rule-based visual language. In *Proceedings of the IEEE Symposium on Visual Languages*, pages 217–224. IEEE, IEEE, September 2000. Seattle USA.

[5] Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[6] Stephen Waydo and Richard M. Murray. Vehicle motion planning using stream functions. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, September 2003.

Syntax-directed Program Visualization

Yoshihiro Adachi
Department of Information and Computer Sciences,
Toyo University
2100 Kujirai, Kawagoe, Saitama 3508585, Japan
adachi@eng.toyo.ac.jp

Abstract - *We developed a syntax-directed approach to program visualization. A class of attribute graph grammars (AGGs) was formalized to define the syntax and layout rules of program diagrams. We developed an attribute evaluation algorithm for the AGG on the basis of its leftmost derivations. The Logichart-AGG was then defined to formalize the syntax and layout rules of a diagram that visualizes a Prolog program and its execution tree in terms of the AGG. All the attribute values of the Logichart-AGG can be evaluated using a simple one-pass attribute evaluation algorithm. We are the first to strictly formalize attribute evaluation algorithms of AGGs based on the traversal of derivation trees in a similar way to the formalization of attribute grammars from context-free string grammars.*

Keywords: program visualization, program diagram, attribute graph grammar, Logichart, graph layout

1 Introduction

We conducted a long-term study on syntax-directed program visualization and developed an *attribute graph grammar* (*AGG*) to formalize the graph-syntactical rules of program diagrams for Pascal, C and Prolog [1], [2]. This AGG consists of a *context-free graph grammar* with formalized productions that specify the syntax rules of the program diagrams and *semantic rules* that are defined so that they can extract the layout information needed to display program diagrams as attributes attached to node labels.

Because the attribute grammar formalism is declarative, the order of evaluation of the attributes must be determined to develop an attribute evaluator. Text strings have a natural linear order from left to right, but the graph nodes generated by the graph grammar do not have a natural linear order. Therefore, developing an attribute evaluation algorithm for an AGG requires a way to determine the order of evaluation of the attributes. Many attribute evaluation methods have been developed for attribute grammars from context-free string grammars, but few have been proposed for AGGs.

We will first define an AGG whose underlying grammar is an edNCE grammar that generates ordered graphs. Next, we will describe attribute evaluation algorithms we developed for the AGG in terms of the node order based on the leftmost derivation. We will then describe an AGG for *Logichart*, a program diagram description language we developed, that formalizes the syntax rules of diagrams that can be used to visualize a Prolog program and its execution tree by using an AGG as a running example. A Prolog visualization system developed based on the Logichart-AGG guarantees that any correct Prolog program can be visualized (*completeness*) and that any Logichart diagram displayed by the system will always be valid for the Logichart-AGG (*soundness*). All attribute values of the Logichart-AGG can be evaluated using a simple one-pass attribute evaluation algorithm.

We are the first to strictly formalize attribute evaluation algorithms of AGGs based on the traversal of derivation trees in a similar way to the formalization of attribute grammars from context-free string grammars.

2 Preliminaries

We describe an edNCE grammar that generates ordered graphs, their leftmost derivations, and derivation trees [3].

2.1 edNCE grammar

Let Σ be an alphabet of nodes and Γ an alphabet of edges. A *graph* over Σ and Γ is a 3-tuple $H = (V, E, \lambda)$, where V is the finite nonempty set of nodes, $E \subseteq \{(v, \gamma, w) \mid v, w \in V, v \neq w, \gamma \in \Gamma\}$ is the set of edges, and $\lambda : V \to \Sigma$ is the node labeling function. The components of a graph H are denoted by V_H, E_H, and λ_H, respectively. Two graphs H and K are *isomorphic* if there is a bijection $f : V_H \to V_K$ such that $E_K = \{(f(v), \gamma, f(w)) \mid (v, \gamma, w) \in E_H\}$ and, for all $v \in V_H$, $\lambda_K(f(v)) = \lambda_H(v)$. The set of all graphs over Σ and Γ is denoted $GR_{\Sigma,\Gamma}$.

Definition 1: A *edNCE grammar* is a tuple $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$, where

(1) Σ is the alphabet of node labels.
(2) Δ is the alphabet of terminal node labels.
(3) Γ is the alphabet of edge labels.
(4) Ω is the alphabet of final edge labels.
(5) P is the finite set of productions. A production is of the form '$X \to (D, C)$', where
    (5.1) $X \in \Sigma - \Delta$ and $D \in GR_{\Sigma,\Gamma}$.
    (5.2) $C \subseteq \Sigma \times \Gamma \times \Gamma \times V_D \times$ {in, out} is the connection relation of p and each element $(\sigma, \beta, \gamma, x, d)$ ($\sigma \in \Sigma$, $\beta, \gamma \in \Gamma$, $x \in V_D, d \in$ {in, out}) of C is a connection instruction of p.

(6) $S \in \Sigma - \Delta$ is the start node label. The *start graph* H_s is a graph that consists of a single node labeled S and no edge. □

To improve readability, a connection instruction $(\sigma, \beta, \gamma, x, d)$ will always be written as $(\sigma, \beta/\gamma, x, d)$. For a production $p : X \rightarrow (D, C)$, X is the left-hand side of p, (D, C) is the right-hand side of p. We write lhs$(p) = X$ and rhs$(p) = (D, C)$.

Two productions $p_1 : X_1 \rightarrow (D_1, C_1)$ and $p_2 : X_2 \rightarrow (D_2, C_2)$ are called isomorphic if $X_1 = X_2$ and there is an isomorphism f from D_1 to D_2 and $C_2 = \{(\sigma,\ \beta/\gamma,\ f(x),\ d) \mid (\sigma,\ \beta/\gamma,\ x,\ d) \in C_1\}$. By copy$(p)$ we denote the set of all productions that are isomorphic to a production p in P. An element of copy(p) is called a *production copy* of p. $copy(P) = \bigcup_{p \in P}$ copy(p).

The process of graph rewriting in an edNCE graph grammar is defined as follows. Let a given graph (host graph) be H and let v be a nonterminal node of H. Let node v be labeled X, and let $p' : X \rightarrow (D', C')$ be a production copy of some $p \in P$ such that D' and H are mutually disjoint.

(1) Remove node v and all edges that are incident with v from H. Let the resulting graph (rest graph) be H^-.
(2) Put D' (daughter graph) in H^-.
(3) Establish new edges between certain nodes of D' and certain nodes of H^- in a way specified by the connection instructions in C'. This is called embedding of D' in H^-. Let the resulting graph be H'.

The meaning of an instruction $(\sigma, \beta/\gamma, x,$ in$) \in C'$ is as follows: if there is an edge with label β from a node $w \in V_H - \{v\}$ with label σ to node v, then the embedding process should establish an edge with label γ from node w to node $x \in V_{H'}$. And similarly for $(\sigma, \beta/\gamma, x,$ out$)$ instead of $(\sigma, \beta/\gamma, x,$ in$)$, where 'in' refers to the incoming edges of v and 'out' to the outgoing edges of v.

Let H and H' be graphs in $GR_{\Sigma,\Gamma}$. If H is transformed into H' by the application of a production copy p' as described above, then we write $H \Rightarrow_{v,p'} H'$ and call it a *derivation step*. A sequence of derivation steps $H_0 \Rightarrow_{v_1,p'_1} H_1 \Rightarrow_{v_2,p'_2} \cdots \Rightarrow_{v_n,p'_n} H_n$ is called a *derivation*. A derivation $H_0 \Rightarrow_{v_1,p'_1} H_1 \Rightarrow_{v_2,p'_2} \cdots \Rightarrow_{v_n,p'_n} H_n$, $n \geq 0$ is *creative* if the graphs H_0 and D'_i, $1 \leq i \leq n$ are mutually disjoint, where rhs$(p'_i) = (D'_i, C'_i)$. We write $H \Rightarrow^* H'$ if there is a creative derivation as above, with $H_0 = H$ and $H_n = H'$. A *sentential form* of G is a graph H such that $H_S \Rightarrow^* H$ for some start graph H_S. A set of sentential forms is denoted by $\mathcal{S}(G)$. The *graph language generated by* G is $\mathcal{L}(G) = \{[H] \mid H \in GR_{\Delta,\Omega}$ and $H_S \Rightarrow^* H$ for some start graph $H_S\}$. For a graph H, the set of all graphs isomorphic to H is denoted $[H]$.

In general, the graph resulting from a derivation based on an edNCE grammar depends on the order in which the production copies were applied. The *confluence* property maintains that the result of a derivation does not depend on the order in which the productions are applied.

Definition 2: An edNCE grammar $G = (\Sigma,\ \Delta,\ \Gamma,\ \Omega,\ P,\ S)$ is *confluent*, or *C-edNCE* grammar, if for all productions $p_1 : X_1 \rightarrow (D_1, C_1)$ and $p_2 : X_2 \rightarrow (D_2, C_2) \in P$, all nodes $x_1 \in V_{D_1}$ and $x_2 \in V_{D_2}$, and all edge labels $\alpha,\ \delta \in \Gamma$, the following equivalence holds:

$$\begin{gathered} \exists \beta \in \Gamma : (X_2, \alpha/\beta, x_1, \text{out}) \in C_1 \quad \text{and} \\ (\lambda_{D_1}(x_1), \beta/\delta, x_2, \text{in}) \in C_2 \\ \Longleftrightarrow \\ \exists \gamma \in \Gamma : (X_1, \alpha/\gamma, x_2, \text{in}) \in C_2 \quad \text{and} \\ (\lambda_{D_2}(x_2), \gamma/\delta, x_1, \text{out}) \in C_1 \end{gathered}$$

□

Proposition 1: An edNCE grammar $G = (\Sigma,\ \Delta,\ \Gamma,\ \Omega,\ P,\ S)$ is confluent if and only if the following holds for every graph $H \in GR_{\Sigma,\Gamma}$:
if $H \Rightarrow_{v_1,p_1} H_1 \Rightarrow_{v_2,p_2} H_{12}$ and $H \Rightarrow_{v_2,p_2} H_2 \Rightarrow_{v_1,p_1} H_{21}$ are (creative) derivations of G, then $H_{12} = H_{21}$. □

2.2 Leftmost derivation

An *ordered graph* is a graph whose nodes have a linear order. Let G be an edNCE grammar such that the right-hand side of its productions are made into ordered graphs by choosing linear orders on their nodes. The right-hand sides of the production copies inherit this order in the obvious way (i.e., they are ordered such that the isomorphism from the original to a production copy is order preserving). This order induces a linear order on the nodes of the sentential forms of G in the obvious way.

The leftmost derivation is defined as follows. Let a given ordered graph (host graph) be H and let v be the first nonterminal node within the node order of H. Let node v be labeled X, and let $p' : X \rightarrow (D', C')$ be a production copy of some $p \in P$ such that D' and H are mutually disjoint.

(1) Remove node v and all edges that are incident with v from H. Let the resulting graph (rest graph) be H^-.
(2) Put D' (daughter graph) in H^-.
(3) Establish new edges between certain nodes of D' and certain nodes of H^- in a way specified by the connection instructions in C'. Let the resulting graph be H'.
(4) If the nodes of H are ordered as $(v_1, \cdots, v_h)$ with $v = v_i$, and those of D' are ordered as $(w_1, \cdots, w_d)$, then the order H' is $v_1, \cdots, v_{(i-1)}, w_1, \cdots, w_d, v_{(i+1)}, \cdots, v_h$.

Let the start graph H_s is an ordered graph that consists of a single node labeled S. If an ordered graph H is transformed into an ordered graph H' by the application of a production copy p' as described above, then we write $H \Rightarrow_{p'} H'$ and call it a *leftmost derivation step*. A sequence of leftmost derivation steps $H_0 \Rightarrow_{v_1,p'_1} H_1 \Rightarrow_{v_2,p'_2} \cdots \Rightarrow_{v_n,p'_n} H_n$ is called a *leftmost derivation*.

We write $H \Rightarrow^*_{lm} H'$ if there is a leftmost derivation from H to H'. The *graph language leftmost generated by* G is $\mathcal{L}_{lm}(G) = \{[H] \mid H \in GR_{\Delta,\Omega}$ and $H_S \Rightarrow^*_{lm} H$ for some start graph $H_S\}$, where the abstract graph $[H]$ does not involve an order any more.

Every derivation of a C-edNCE grammar can be transformed into a leftmost derivation, by repeated application of the confluence property.

Proposition 2: For every C-edNCE grammar G, $\mathcal{L}_{lm}(G) = \mathcal{L}(G)$.

2.3 Derivation tree

Derivation trees are rooted, ordered trees. Each vertex of the tree has directed edges to each of its k children, $k \geq 0$, and the order of the children is indicated by labeling the edges as $1, \cdots, k$.

Definition 3: Let $G = (\Sigma,\ \Delta,\ \Gamma,\ \Omega,\ P,\ S)$ be an edNCE grammar. A *c-labeled derivation tree* of G is a rooted, ordered tree t of which the vertices are labeled by production copies in $copy(P)$, such that

(1) the right-hand sides of all the production copies that label the vertices of t are mutually disjoint, and do not contain the root of t as a node, and
(2) if vertex v of t has label $X \to (D, C)$, then the children of v are the nonterminal nodes of D, and their order in t is the same as their order in D; moreover, for each child w, the left-hand side of the label of w in t equals its label in D. □

For a c-labeled derivation tree t, let $v_1, v_2, \cdots, v_n$ be the sequence of its vertices in pre-order, let p_i be the label of v_i, and let $X = \text{lhs}(p_1)$. Then, $H_0 \Rightarrow_{v_1,p_1} H_1 \Rightarrow_{v_2,p_2} \cdots \Rightarrow_{v_n,p_n} H_n$, where H_0 consists of a single node v_1 labeled X and H_n is terminal graph, is a leftmost derivation.

3 AGG and attribute evaluation

We first define an AGG whose underlying grammar is an edNCE grammar that generates ordered graphs. We then define our attribute evaluation algorithm for the AGGs.

3.1 Definition of AGG

An AGG in this paper consists of an underlying edNCE grammar that generates ordered graphs, together with attributes and semantic rules defining the values of these attributes.

Definition 4: An *attribute graph grammar* (*AGG*) is a tuple $AG = < G,\ A,\ F >$, where

(1) $G = (\Sigma, \Delta, \Gamma, \Omega, P, S)$ is edNCE grammar that generates order graphs and we call it an *underlying graph grammar* of AG.
(2) Each node label $X \in \Sigma$ of G has two disjoint finite sets $I(X)$ and $S(X)$ of *inherited* and *synthesized attributes*, respectively. We denote the set of all attributes of nonterminal node symbol X by $A(X) = I(X) \cup S(X)$. $A = \bigcup_{X \in \Sigma} A(X)$ is called the *set of attributes* of G. We assume that $I(S) = \phi$, and that for each $X \in \Delta$ $S(X) = \phi$. An attribute of X is denoted by $a(X)$, and the set of possible values of a is denoted by $\mathcal{V}(a)$.
(3) Associated with each production $p : X_0 \to (D, C)$ is a set F_p of *semantic rules*, which define all the attributes in $S(X_0) \cup I(X_1) \cup \cdots \cup I(X_n)$ where $V_D = \{v_1, \cdots, v_n\}$ and $\lambda_D(v_i) = X_i, 1 \leq i \leq n$. A semantic rule defining an attribute $a(X_k), 0 \leq k \leq n$ has the form $a(X_k) := f(a_1(X_{i_1}), \ldots, a_m(X_{i_m}))$, where f is a mapping $\mathcal{V}(a_1(X_{i_1})) \times \ldots \times \mathcal{V}(a_m(X_{i_m}))$ into $\mathcal{V}(a(X_k))$. In this situation, we say that $a(X_k)$ depends on $a_1(X_{i_1}), \cdots, a_m(X_{i_m})$ in p. The set $F = \bigcup_{n \in P}$ is called the *set of semantic rules* of AG. □

3.2 Attribute evaluation algorithm

As previously mentioned, graph nodes generated by AGGs have no natural linear ordering. Therefore, developing an AGG attribute evaluation algorithm requires a way to assign an order to traversing vertices of c-derivation trees of the AGG to evaluate its attributes. We use an edNCE grammar that generates ordered graphs as the underlying grammar of the AGG. The vertices of c-derivation trees are then traversed in the order within nodes of the right-hand side graphs of the edNCE grammar's productions.

A simple one-pass (l-pass) attribute evaluation algorithm (*simple one-pass attribute evaluator*) for AGGs is defined in Figure 1. It traverses vertices of a c-derivation tree t from top to bottom and left to right. An AGG G is *simple one-pass* if the simple one-pass evaluator evaluates all attribute instances of every c-derivation tree of G.

```
procedure EVALUATE(v0); vertex v0;
{Let the label of vertex v0 be X0 → (D, C) and let the nodes
of D be ordered as v1 v2 ··· vn.}
  begin
    if X0 ≠ S then compute all inherited attributes of X0;
    for i := 1 to n do
      begin
        if vi is a nonterminal node then EVALUATE(vi);
        else compute all attributes of λD(vi);
      end
    compute all synthesized attributes of X0;
  end

{main program}
begin
  EVALUATE(the root of t);
end
```

Fig. 1. Simple one-pass evaluator

Pure (simple) visit, sweep, alt-pass, and pass attribute evaluation procedures for attribute grammars were discussed in [4]. Formalizing them is straightforward for the AGGs on the basis of the order within nodes of the right-hand side graphs of the edNCE grammar's productions and their c-derivation trees.

4 Prolog program visualization

We describe the Logichart-AGG that formalizes the syntax and layout rules of diagrams that can be used to visualize a Prolog program and its execution tree by using an AGG as a running example.

4.1 Logichart diagram

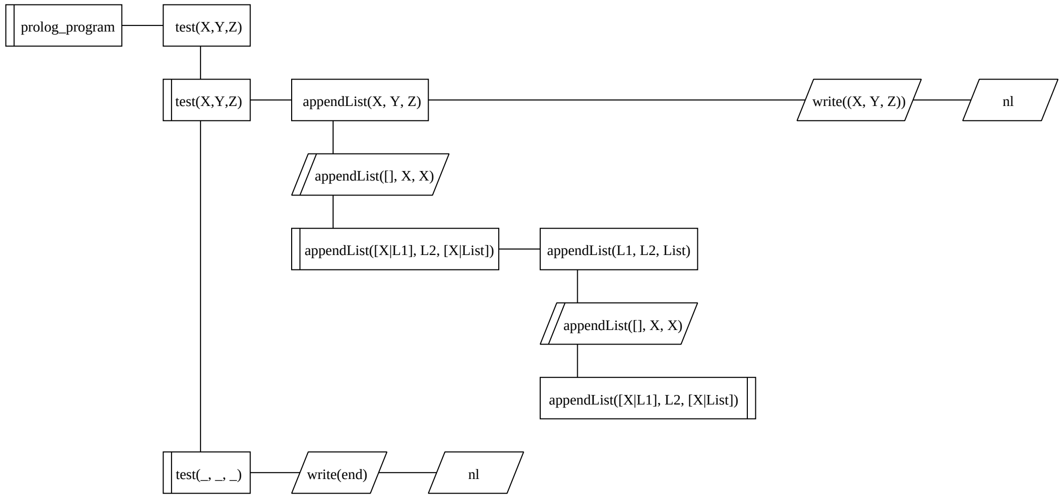


Fig. 2. An example of Logichart diagrams

Logichart, which is short for Logic flowchart, is a program diagram description language that we have developed to visualize Prolog programs and their execution flows[2], [5]. Prolog programs and their executions are visualized by using a Logichart diagram as follows. If a query '?- $G_1, G_2, \cdots, G_n$' is input, the clause 'prolog_program :- $G_1, G_2, \cdots, G_n$' is added to the Prolog program. The head of this clause is visualized as a node labeled with the string 'prolog_program' and becomes the root of the Logichart diagram. The Logichart diagram that corresponds to the Prolog program below and the query '?- test.' is shown in Figure 2.

```
test(X,Y,Z) :- appendList(X,Y,Z),
     write((X,Y,Z)),nl.
test(_,_,_) :- write(end),nl.
appendList([],X,X).
appendList([X|L1],L2,[X|List]) :-
     appendList(L1,L2,List).
```

A Logichart diagram has a tree-like structure as shown in Figure 2 with the following features: the head and body goals that compose each clause are aligned horizontally and calling goals and the heads of the clauses that they call are aligned vertically. The first feature gives clauses in a Prolog program and their representations in the corresponding Logichart diagram a similar structure, so the correspondences between them can more easily be seen. The second feature makes relationships between clauses easier to grasp because such clauses are located on the same vertical line. Note that a Logichart diagram includes an execution subtree whose root is 'prolog_program'.

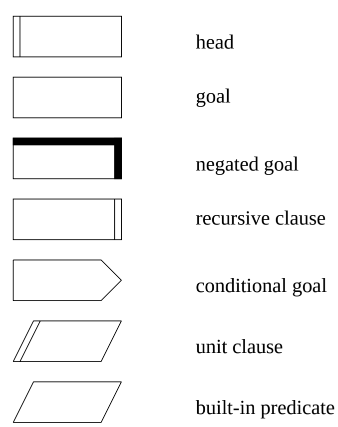
4.2 Logichart-AGG

The Logichart-AGG consists of an edNCE graph grammar that generates ordered graphs and semantic rules. The edNCE grammar has formalized productions that can specify the syntax rules of Logichart diagrams, and the semantic rules enable calculating the node coordinates needed to display Logichart diagrams as attributes attached to node labels. The Logichart-AGG specifications consist of 17 productions associated with 264 semantic rules.

The node label alphabet of the Logichart-AGG is shown in Figure 3. The edge label alphabet is $\Gamma = \Omega = \{e, h, v\}$.

[prolog_program]	start label
[clause-sequence]	clause sequence
[clause]	clause
[goal]	goal

(a) The nonterminal node alphabet



(b) The terminal node alphabet

Fig. 3. The node alphabets of the Logichart-AGG

The following attributes are defined to extract node coordinates needed to display Logichart diagrams.
Inherited attributes of node labels (except the start label):

- π_x : the x-coordinate of a node
- π_y : the y-coordinate of a node

Synthesized attributes of the nonterminal node labels:

- subtree_width : the width of a subtree
- subtree_depth : the depth of a subtree

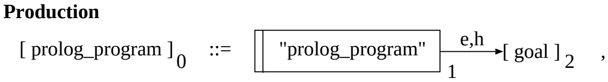
The following symbols and functions are used in the productions and semantic rules of the Logichart-AGG.

- <X> : the text string corresponding to the Prolog goal name X
- "X" : text string X
- max(X,Y) : a function returns the maximum value of X and Y
- get_width(X) : a function returns the node width obtained from text string X
- get_depth(X) : a function returns the node depth obtained from text string X

Some of the productions and their associated semantic rules in the Logichart-AGG are outlined below. The number written in the lower right of each node of the productions is its identifier, and denotes its order within the nodes of the right-hand side graphs.

(a) The production and semantic rules to rewrite the start node '[prolog_program]' are shown in Figure 4. These rules are formalized to represent queries given in Prolog syntax, and the nonterminal node 'goal_sequence' in the right-hand-side graph corresponds to the query. A graph that is isomorphic to the right-hand side of Production 1 is derived by applying this production to the start node. Semantic rules $\pi_x(1) = \mathrm{RootX}$ and $\pi_y(1) = \mathrm{RootY}$ mean that the x-coordinate of node '1' is 'RootX' and the y-coordinate of node '1' is 'RootY'. Semantic rule $\pi_x(2) = \mathrm{RootX} + \mathrm{get\_width}(\text{"prolog\_program"}) + \mathrm{GapX}$ means that the x-coordinate of node '2' is equal to 'RootX' plus the width of the node labeled "prolog_program" plus the horizontal gap 'GapX'. Semantic rule $\pi_y(2) = \mathrm{RootY}$ means that the y-coordinate of node '2' is 'RootY'. The root node "prolog_program" and the subdiagram derived from the nonterminal node '2' labeled 'goal' are aligned with a horizontal separation of 'GapX' by these semantic rules.

Production



C = ∅

Semantic Rules

$\pi_x(1)$ = RootX
$\pi_y(1)$ = RootY
subtree_width(0) = get_width("prolog_program") + GapX + subtree_width(2)
subtree_depth(0) = max(get_depth("prolog_program"), subtree_depth(2))
$\pi_x(2)$ = RootX + get_width("prolog_program") + GapX
$\pi_y(2)$ = RootY

Fig. 4. Rules to rewrite the start node '[prolog_program]'

(b) The production and semantic rules shown in Figure 5 are as formalized for the 'and' operation on Prolog goals. A node labeled 'goal' is replaced with a graph that is isomorphic to the right-hand side of Production 2 by applying this production. Semantic rule $\pi_x(2) = \pi_x(1) + \mathrm{subtree\_width}(1) + \mathrm{GapX}$ means that the x-coordinate of node '2' is equal to the x-coordinate of node '1' plus the width of the subdiagram derived from node '1' plus the horizontal gap 'GapX'. Therefore, goals connected by the operator 'and' are aligned with a separation of 'GapX' in the horizontal direction.

Production

C = {(#,e,e,1,in), (#,e,e,2,out), (#,h,h,1,in), (#,h,h,2,out), (#,v,v,1,in), (#,v,v,1,out)}

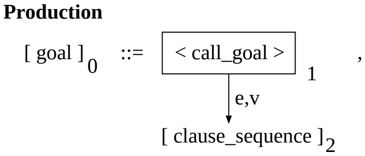
Semantic Rules

$\pi_x(1) = \pi_x(0)$
$\pi_y(1) = \pi_y(0)$
subtree_width(0) = subtree_width(1) + GapX + subtree_width(2)
subtree_depth(0) = max(subtree_depth(1), subtree_depth(2))
$\pi_x(2) = \pi_x(0)$ + subtree_width(1) + GapX
$\pi_y(2) = \pi_y(0)$

Fig. 5. Rules formalized for the 'and' operation on Prolog goals

(c) The production and semantic rules shown in Figure 6 are formalized for the call of a goal. The semantic rule $\pi_y(2) = \pi_y(1) + \mathrm{depth}(1) + \mathrm{GapY}$ means that the y-coordinate of node '2' is equal to the y-coordinate of node '1' plus the depth of the cell corresponding to node '1' plus the vertical gap 'GapY'. Therefore, a calling goal and the clause heads of the goals called by it are aligned with a separation of 'GapY' in the vertical direction.

Production



C = {(#,e,e,1,in), (#,e,e,1,out), (#,h,h,1,in), (#,h,h,1,out), (#,v,v,1,in), (#,v,v,2,out)}

Semantic Rules

$\pi_x(1) = \pi_x(0)$
$\pi_y(1) = \pi_y(0)$
subtree_width(0) = max(get_width(<call_goal>), subtree_width(2))
subtree_depth(0) = get_depth(<call_goal>) + GapY + subtree_depth(2)
$\pi_x(2) = \pi_x(1)$
$\pi_y(2) = \pi_y(1)$ + get_depth(1) + GapY

Fig. 6. Rules formalized for the call of a goal

(d) The production and semantic rules used to rewrite a nonterminal node '[goal]' with a terminal node that corresponds to the system predicates such as 'write', 'nl', and the cut '!' are shown in Figure 7.

It is straightforward to show that the underlying edNCE grammar of the Logichart-AGG satisfies the conditions described in Definition 2.

Production

$$[\text{ goal }]_0 \;::=\; \langle\text{ system\_predicate }\rangle_1 \quad ,$$

$$C = \{(\#,e,e,1,in), (\#,e,e,1,out), (\#,h,h,1,in), (\#,h,h,1,out), (\#,v,v,1,in), (\#,v,v,1,out)\}$$

Semantic Rules

$\pi_x(1) = \pi_x(0)$
$\pi_y(1) = \pi_y(0)$
subtree_width(0) = get_width(< system_predicate >)
subtree_depth(0) = get_depth(<system_predicate>)

Fig. 7. Rules to rewrite a nonterminal node '[goal]' with '⟨ system_predicate ⟩'

Theorem 1: The underlying edNCE grammar of the Logichart-AGG is confluent. □

The confluent property guarantees that every derivation of the Logichart-AGG can be transformed into a leftmost derivation without changing the graph derived by it.

It is straightforward to show that the simple one-pass evaluator described in Figure 1 evaluates all attribute instances of every c-derivation tree of the Logichart-AGG.

Theorem 2: The Logichart-AGG is simple one-pass. □

The fact that the Logichart-AGG is formalized in complete accord with the Prolog syntax guarantees that a Logichart can be drawn for any correct Prolog program (completeness) and that any Logichart displayed by it is always valid for the Logichart-AGG (soundness).

Claim 1: The set of Logichart diagrams generated by the Logichart-AGG is equivalent to the set of syntactically correct Prolog programs. □

5 Prolog visualization system

We have developed a Prolog visualization system completely based on the Logichart-AGG. It consists of a translator, viewer, and tracer. The input to it is a Prolog program and a goal (query) entered via the graphical user interface.

Translator From the input Prolog program and query, the translator generates a symbolic representation of the Logichart diagram composed of the clauses that are successively called by the query execution.

Viewer The viewer takes as its input the symbolic representation of a Logichart diagram and displays it in an X-Window, as shown in Figure 8.

Tracer The tracer is a meta-interpreter that displays the goal execution process in the Logichart diagram; the color of each cell changes depending on the result (true or false) of the corresponding goal execution, and the change in each variable is displayed in real time, as shown in Figure 9.

6 Conclusion

We have discussed program visualization based on attribute graph grammars. Syntax-directed formalization of program diagrams by using the AGG is applicable not only

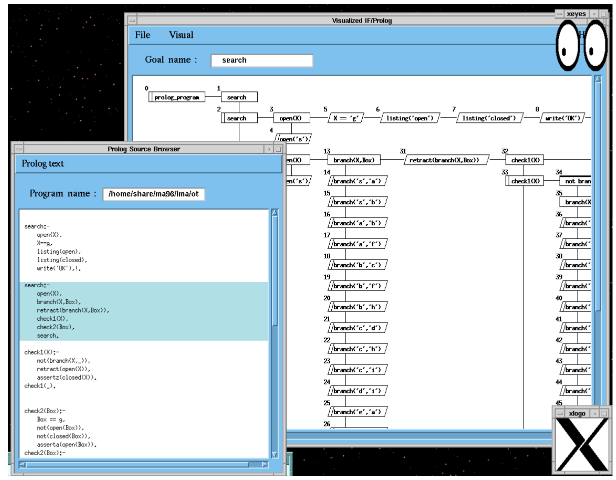


Fig. 8. Screen displaying a Logichart diagram

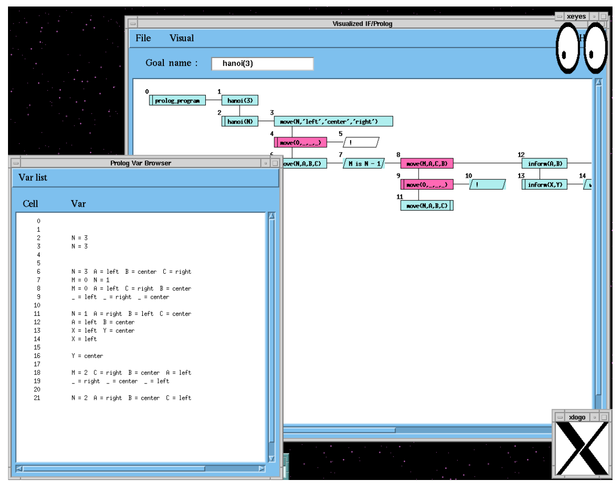


Fig. 9. Trace of Prolog execution

to Prolog but also to many other programming languages, including C, Pascal, and Java. It will provide a sophisticated and practical foundation for visualizing programs and for visual programming systems.

References

[1] Y. Adachi, K. Anzai, K. Tsuchida and T. Yaku, Hierarchical Programming Diagram Editor Based on Attribute Graph Grammar, *Proceedings of the IEEE 20th Annual International Computer Software and Applications Conference* (1996), 205-213.

[2] Y. Adachi, T. Imaki, K. Tsuchida and T. Yaku, Prolog Visualization Based on Attribute Graph Grammar, *Proceedings CDROM of the Fundamental Conference of the 15th IFIP World Computer Congress* (1998).

[3] J. Engelfriet and G. Rozenberg, Node Replacement Graph Grammar, in *Handbook of Graph Grammars and Computing by Graph Transformation* (Rozenberg, G., eds), World Scientific (1997), 1-94.

[4] J. Engelfriet and F. Gilberto, Passes, Sweeps, and Visits in Attribute Grammars, Journal of the Association for Computing Machinery, Vol.36, No.4 (1989), 841-869.

[5] Y. Adachi, K. Tsuchida, T. Imaki and T. Yaku, Logichart - Intelligible Program Diagram for Prolog and its Processing System, *Electronic Notes in Theoretical Computer Science*, Volume 30, Issue 4, Elsevier Science (2000).

Staying Oriented with Software Terrain Maps

Robert DeLine
Microsoft Research, Microsoft Corporation, Redmond, USA

Abstract

Developers often find themselves lost as they navigate around large programs, particularly when those programs are unfamiliar. This paper presents a new visualization, called a software terrain map, intended to keep a programmer oriented as she navigates around source code in the editor. The design is based on the metaphor of cartographic maps, which are continuous (no wasted space), have enough visual landmarks to allow the user to find her location perceptually rather than cognitively, and lend themselves to overlaying data. Although an optimal layout for software terrain maps is computationally intractable, the paper presents an efficient, heuristic algorithm that produces good results.

1 Introduction

In our recent study in which experienced developers attempted to enhance an unfamiliar program, the participants consistently got lost in the source code. [1] They typically explored the code by opening and scanning many documents, by jumping back and forth among related definitions, and by iterating over the result sets of text searches. Despite their years of experience and their familiarity with the programming language, development environment, and problem domain, these navigation steps would quickly leave them disoriented. Many times, a participant would issue a new query to find a previously visited definition, and in a few cases, a participant would even inspect a previously visited definition and not recognize it.

Such disorientation is easy to understand. A typical program is very uniform in visual appearance and relies heavily on names to distinguish its parts. A typical development environment, like Visual Studio or Eclipse, also relies on names in its overview displays, like the tree of project files and the type hierarchy. Hence, staying oriented while navigating a program requires familiarity with its names, which places a large burden on both short- and long-term memory. In this paper, I describe a new display, intended to allow the programmer to use her spatial memory to stay oriented while navigating the source code.

To keep the user oriented during code navigation, I propose supplementing the development environment with an overview diagram to show the programmer's current location in the program ("you are here") and recent navigation steps (a vapor trail). In addition to reflecting the user's navigation in the editor, the overview would also allow navigation. For instance, clicking on the overview would also cause the editor to show the corresponding part of the code. The intent is to allow the programmer to use spatial memory to navigate to sought parts of the program. To realize this intent, several desiderata seem reasonable:

1. The display should show the entire program. That is, whatever definition the programmer navigates to in the editor should have a representation at a reasonable level of detail in the display. Hence, the use of elision or abstraction to scale to large programs would not be appropriate. Elision would cause some navigation steps to be "off the map," while abstraction would cause a navigation step within an abstracted part of the program to appear as non-movement in the overview display.
2. The display should contain enough visual landmarks to allow the developer to find parts of the program perceptually, rather than relying on names or other cognitive cues. For instance, to find Rome on a map of Europe, I scan for Italy's famous boot shape rather than reading for the word *Italy*. An overview display of software should have similar visual landmarks.
3. The display should remain visually stable as the user navigates. If the display's content were to change as much as the editor's while the user navigates around the program text, it would provide little help in keeping the user oriented. Further, editing the program text should cause proportional changes to the display.
4. Finally, to justify the additional screen space needed for the display, the display should be capable of showing global program information other than navigation steps. For instance, we might use the display to show program execution paths, like the call stack when an exception is raised or the hot path that a profiler reports, or to show team awareness data, like which developers are currently working on which parts of the program.

These desiderata mean that several popular technologies are not suitable for such a display. UML class diagram

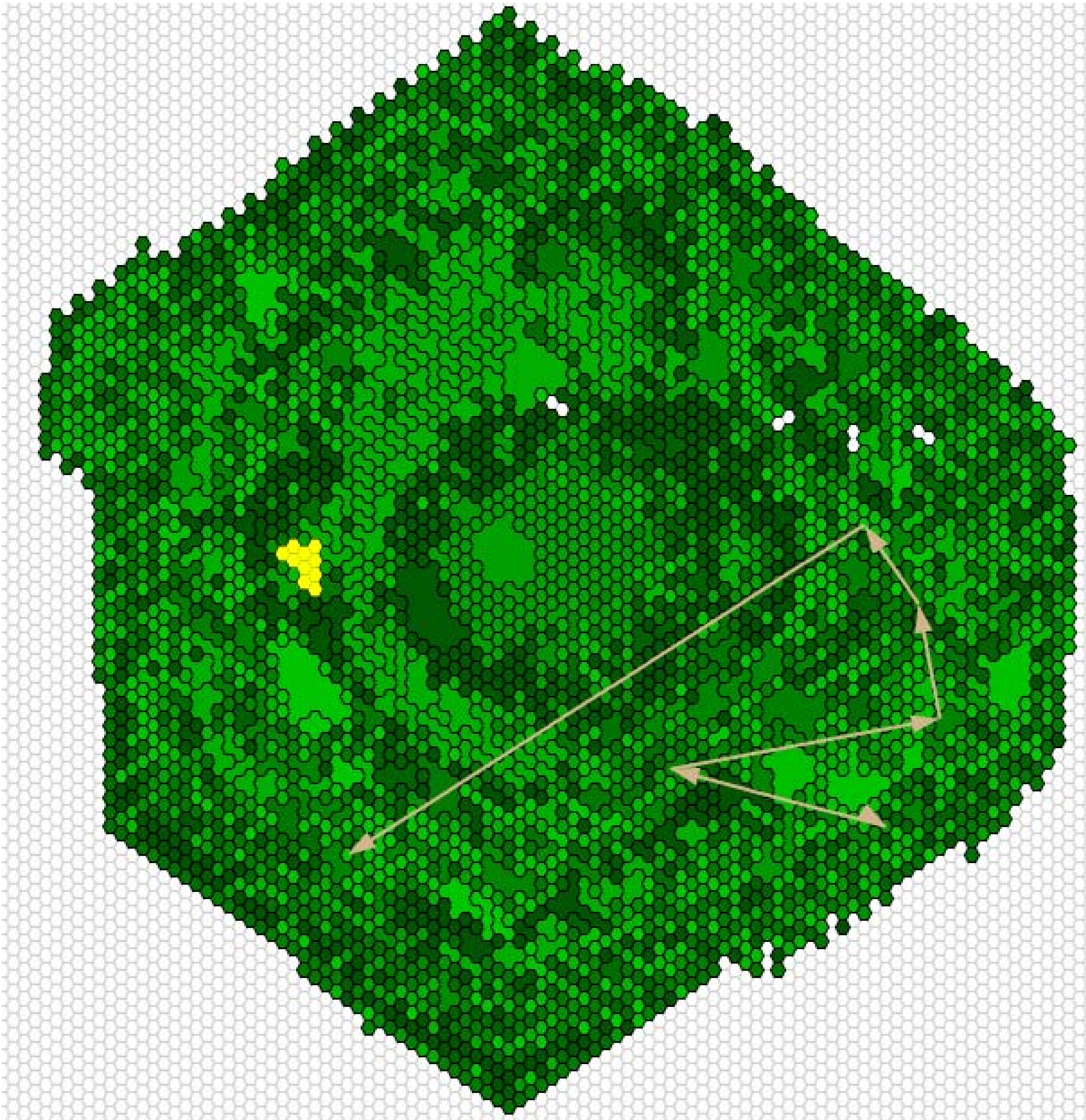


Figure 1. The software terrain map of a library for analyzing .NET bytecode. Each of the 3800 regions is a method, whose size is proportional to the method's textual size (one tile for every two lines of code). Each class has its own shade of green. The highlighted (yellow) method is the one that the user is currently reading in the editor. The arrows show the program's execution path up to the current debugger breakpoint.

and other box-and-line architectural diagrams, for instance, are not good candidates. First, they are often drawn at an inappropriately high level of abstraction or elide parts of the program to keep the diagrams small. A developer working on an object-oriented program navigates among and edits individual class members. Hence, to provide a location marker while remaining visually stable, the display must show every member in the program all at once. This would be difficult to do with a UML class diagram. Second, such diagrams are visually uniform, particularly when they include many boxes. Distinguishing the boxes is more easily accomplished

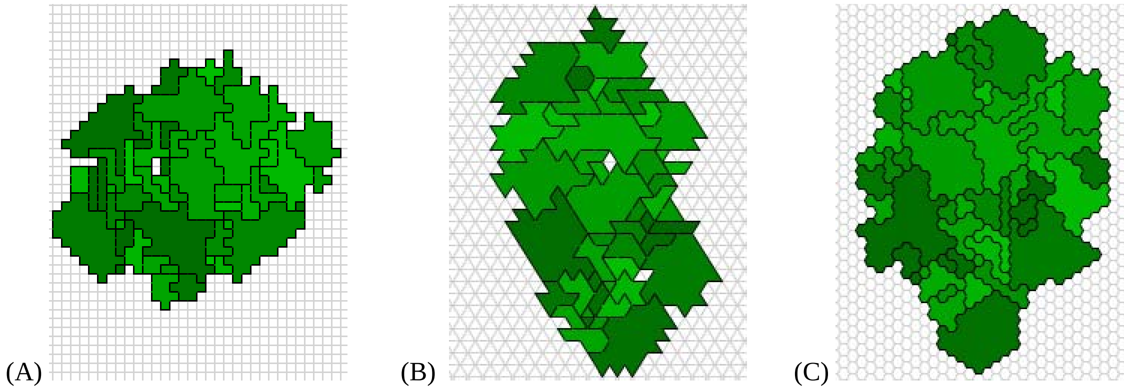


Figure 2. Three software terrain maps for the same library, built on three grid systems: (A) squares; (B) equilateral triangles; and (C) hexagons.

by reading their labels than by relying on visual cues like box position or the topology of edges between boxes.

A popular technique for scaling complex information to fit the screen is to use a detail-within-context display, like fisheye views.[2] Such a display gives more screen space and provides more detail about the user's current focus, diminishing and abstracting the other parts of the information. For instance, SHriMP uses this technique on box-and-line displays of software structure.[4] Using this technique for a navigation display would mean that the display would change every time the developer moved in the text editor. The use of animated transitions diminishes the disorientation caused by changing the display, but only when the user's attention is on the animation (and not on the editor, for instance). Even if the navigation display and editor occupy the same window, changing both simultaneously is confusing and unlikely to keep the user oriented.

2 Software Terrain Maps

To satisfy the desiderata above, I am designing a new software visualization called a *software terrain map*, based on the metaphor of cartographic maps. An example is shown in Figure 1. Cartographic maps have many nice properties: they are continuous (no wasted space) and stable (except perhaps at a geological time scale); they contain obvious and memorable visual landmarks (e.g. the shapes of boundaries, the positions of natural features); they lend themselves to overlaying data, both for easing navigation (e.g. names, roads, icons for features) and for conveying information in context (e.g. political, demographic, or economic patterns); and they are very familiar. A software terrain map is designed to show all of a software's parts, either behind the text in the editor window or on a second monitor. A highlight on the map continuously updates to show the part that the programmer is currently editing.

To mimic the continuous nature of cartographic maps, I partition the screen into tiles and assign tiles to the software parts. An algorithmically inexpensive approach is to choose locations for the parts and then to draw a Voronoi diagram around the locations to partition the screen. However, the shapes of the tiles constitute the display's major visual landmarks, and the use of Voronoi diagrams provides only indirect control over the tile shapes. Instead, I first partition the screen into regularly shaped tiles and then assign tiles to the software parts. As can be seen in Figure 1, the result gives the map an overall tidy, regular appearance, while containing enough irregularities to create visual landmarks.

Mathematicians, beginning with Golomb in the 1950s, have studied building shapes out of regularly tilings of the plane.[3] In particular, they have studied building shapes from squares, which they call *polyominoes* (see Figure 2A); from triangles, which they call *polyiamonds* (see Figure 2B); and from hexagons, which they call *polyhexes* (see Figure 2C). Software terrain maps can be drawn based on any of these three.

2.1 Layout Problem

To draw a software terrain map, we model the program as a set of *components* described by two metrics, which are parameters to the layout algorithm: Size(c) which gives the number of contiguous tiles to assign to the component c; and Affinity(c_1, c_2), which is the degree to which components c_1 and c_2 are related. The problem, then, is to locate components near each other in proportion to their affinity, while assigning each component the appropriate number of tiles. That is, computing the layout is an constrained optimization problem to find

$$\min \forall(c_1, c_2) \bullet \text{Affinity}(c_1, c_2) \times \text{Distance}(c_1, c_2)$$

where Distance is a measure of screen distance between components, for example the Euclidean distance between their centroids.

This formulation of the layout problem is intentionally generic to allow me to explore various useful notions of size and affinity. For the terrain map in Figure

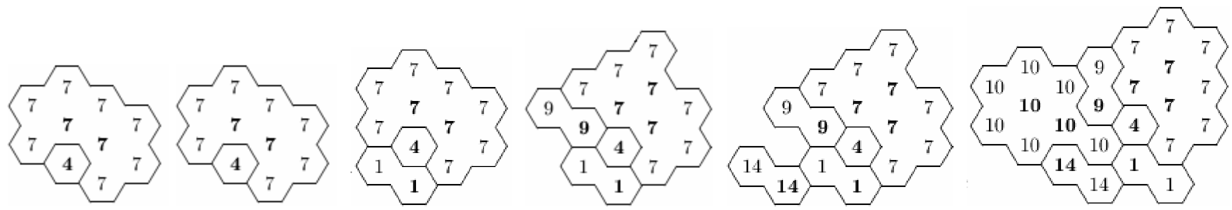


Figure 3. The first six steps of the layout algorithm (left to right), showing the addition of components identified by numbers: component #7 has size 9; #4, size 1; #1, size 2; #9, size 2; #14, size 2; and #10, size 8. Each of a component's coordinates is labeled with the component's number; taken coordinates are in boldface.

1, the components are methods and the number of tiles assigned to a method is proportional to its text size (in this case, one tile for every two lines of code). The intention is that methods that appear large in the editor also appear large in the terrain map, helping the programmer to learn the correspondence between the two.

For this figure, affinity is an arbitrary score chosen to reflect both control and data flow. In this case, to compute the affinity between c_1 and c_2, I give 20 points if c_1 calls c_2, 20 additional points if c_2 calls c_1, and one point for each field read or written in both c_1 and c_2. There is nothing "magical" about this formula. Part of the research agenda is to find formulas for affinity that have two nice properties: (1) the more often the developer navigates between two methods, the closer they appear on screen; (2) paths overlaid on the map (such as execution paths) are drawn as "snakes" rather than "scribbles." Ideally, a formula for affinity would involve information only about the program's static structure (since this is immediately available), but it may also be useful to include measures from the program's source history, from traces of its execution behavior, or even from the team's bug database or communications.

2.2 Layout Algorithm

The layout problem, while conveniently generic, is also computationally intractable, with clear relationships to both bin packing and the traveling salesman problem. Fortunately, there is a heuristic quadratic algorithm that produces appealing results. (Here, quadratic time is a lower bound for this problem since inspecting the N×N affinity matrix alone requires quadratic time. The bound could be lowered further by disregarding large portions of the affinity matrix.) The gist of the algorithm is to add each component to the map one at a time, spiraling outward from the center, and to adjust the shape of previously added components to improve their proximities to high-affinity components. This shape adjustment is limited to keep the overall algorithm quadratic.

The algorithm proceeds in two phases. First, we choose an order in which to add the components to the map. This order, perhaps even more than the size and affinity metrics, affects the map's final appearance, so I am experimenting with several choices. The order used to produce Figure 1 first sorts the methods by each method's total affinity for other methods. For each method *m* in this sorted list, *m* is added to the final ordering, then we add all those methods reachable from *m* through a depth-first traversal of the program's call graph. At each stage of the depth-first traversal, the callees are explored in the order from highest to lowest total affinity. Two other approaches are to use a breadth-first search of the call graph and to ignore the call graph altogether and simply use the list sorted by total affinity. Of these three, the depth-first approach produces execution path overlays that are less "scribbly" than the other two approaches, at least in my initial experiments.

The second phase of the algorithm adds the components to the map in the order that the first phase determines. For each component to be added, we first compute the previously added component with the highest affinity for the new component, which I call the target. The second phase attempts to get the new component as close as possible to the target, without using more than linear time to find a good position. The second phase of the algorithm is parameterized by the grid to be used, namely, one of the three shown in Figure 2. A grid, as an abstract data type, supports a single operation: CoordinatesAtDistance(d, (x,y)) returns the set of grid coordinates at Manhattan distance d from coordinate (x,y). I'll use the expression Neighbors(x,y) to mean CoordinatesAtDistance (1, (x,y)).

To allow a component's shape to be adjusted as new components are added, each component is assigned two types of coordinates: a component's *taken* coordinates are fixed (i.e. the taken coordinate belongs to the component now and forever); a component's *claimed* coordinates can be exchanged for other coordinates. The algorithm maintains the following invariants:

(1) the number of a component's taken and claimed coordinates equals the component's size, i.e. Taken(c) + Claimed(c) = Size(c);

(2) the component's taken coordinates are all contiguous, i.e. $\forall(x,y) \in \text{Taken}(c) \bullet \exists\, (x',y') \in \text{Taken}(c) \bullet (x,y) \in \text{Neighbors}\,(x',y')$; and

(3) every claimed coordinate is the neighbor of a taken coordinate, i.e. $\forall(x,y) \in \text{Claimed}(c) \bullet \exists\ (x',y') \in \text{Taken}(c) \bullet (x,y) \in \text{Neighbors}\ (x',y')$.

Subject to these invariants, each component maintains as few taken coordinates as possible, since the more claimed coordinates a component has, the more flexible its shape, due to a process called claim renouncing, described below.

Pseudocode for the core of the algorithm is shown at right. For each component to be added, the algorithm begins looking at distance 1 from the target and proceeds to greater distances until enough room for the new component has been found. For a given distance, we first divide the coordinates at that distance into the empty ones (the ones that no component has claimed or taken) and the claimed ones. We first consider each empty space in turn as a possible root for adding the new component, turning to the claimed ones only if there are no suitable empty ones. At each root, we search for enough coordinates to assign to the component to make up its size. If we cannot find enough coordinates at that root, any state changes made to the grid are abandoned and we try the next candidate root.

To search for coordinates from a root coordinate, the component first takes the root coordinate. To find each additional coordinate needed, we search among the direct neighbors (coordinates at distance 1) of the component's taken and claimed coordinates. The component can grow to include a neighboring coordinate either if the coordinate is empty or if the coordinate is claimed by another component willing to renounce its claim (as described below). When the component finds a candidate neighboring coordinate, it claims it. If this newly claimed coordinate is a direct neighbor of one of the component's taken coordinates, the invariants are maintained and the algorithm can continue the search. However, if the newly claimed coordinate is a direct neighbor only of the component's claimed coordinates, then invariant (3) is violated. To re-establish the invariant, we convert one of the claimed coordinates to a taken coordinate and then continue the search.

For a component to renounce its claim on a coordinate, it must find a replacement coordinate to claim instead. The search for the replacement is exactly as described in the previous paragraph, with two exceptions. First, we must keep track of the coordinate being renounced so that the search for a replacement does not end up finding the one we want to renounce. In fact, since the search for a replacement can cause neighboring components to try to renounce their own claims, we must track all coordinates being renounced. (Otherwise, we can get cycles of neighboring components fruitlessly swapping renounced coordinates.) Finally, this recursive process of neighbors renouncing claimed is limited by a

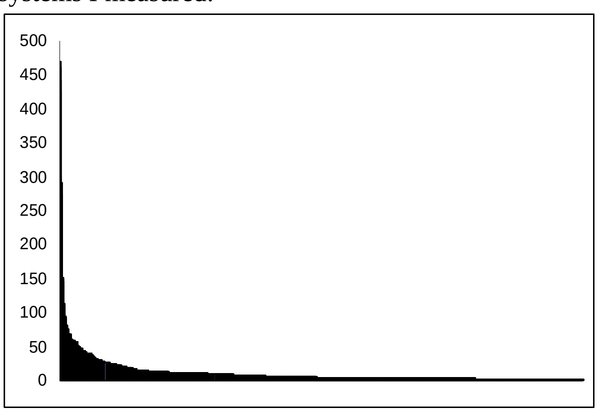
```
type Component :
  var grid : Grid;
  var layout : GridLayout;

  PlaceNear (target : Component) :
    for distance := 1 to ∞ :
      candidates := layout.CoordsAtDistance(distance, target);
      foreach coord in EmptyCoordinates(candidates, grid) :
        if this.PlaceAt(coord) : return;
      foreach coord in ClaimedCoordinates(candidates, grid) :
        if this.PlaceAt(coord) : return;
  end PlaceNear

  PlaceAt (start: Coord) : bool
    grid.BeginTransaction();
    if grid.Claimed(start) ∧ ¬ grid.Claimant(start).Renounce(start,{},MAX)
      grid.AbortTransaction()
      return false;
    grid.Take(this, start);
    placesToExpand := new Queue<Coord>;
    placesToExpand.Enqueue(start);
    while ¬ this.CompletelyInGrid(grid) ∧ placesToExpand.Count > 0
      placeToExpand := placesToExpand.Dequeue();
      expanded := false;
      foreach c in grid.Neighbors(placeToExpand)
        if grid.IsEmpty(c) ∨
          grid.IsClaimed(c) ∧ grid.Claimant(c).Renounce(c, {c}, MAX)
          expanded := true;
          grid.Claim(c, this);
          placesToExpand.Enqueue(c);
      if expanded ∧ this = grid.Claimant(placeToExpand)
        grid.Take (placeToExpand, this);
    if this.CompletelyInGrid(grid)
      grid.CommitTransaction();
      return true;
    else
      grid.AbortTransaction();
      return false;
  end Place

  Renounce (toRenounce: Coord, forbidden: Set<Coord>, limit: int) : bool
    if limit = 0 : return false;
    othersClaims := {};
    foreach takenCoord in grid.TakenSet(this)
      foreach c in layout.Neighbors(takenCoord) \ forbidden
        if grid.IsEmpty(c)
          grid.Unclaim(toRenounce, this);
          grid.Claim(c, this);
          return true;
        else if grid.IsClaimed(c) ∧ this ≠ grid.Claimant(c)
          othersClaims.Add(c);
    foreach c in othersClaims
      if grid.Claimant(c).Renounce(c, forbidden ∪ {c}, limit-1)
        grid.Unclaim(toRenounce, this);
        grid.Claim(c, this);
        return true;
    foreach cl in grid.Claimed(this) \ forbidden
      foreach c in layout.Neighbors(cl) \ forbidden
        if grid.IsEmpty(c) ∨
          grid.IsClaimed(c) ∧
            grid.Claimant(c).Renounce(c, forbidden ∪ {c,cl}, limit-1)
          grid.Take(cl, this);
          grid.Unclaim(toRenounce, this);
          grid.Claim(c, this);
          return true;
    return false;
  end Renounce
end Component

type Terrain :
  SolveLayout (comps: List<Component>, layout: GridLayout) :
    target := new Map<Component,Component>;
    for i := 0 to size(comps)-1 :
      target[comps[i]] := CompWithMaxAffinity(Sublist(comps,0,i));
    grid := new GridState
    foreach comp in comps :
      comp.layout := layout;
      comp.grid := grid;
      comp.PlaceNear(target[comp]);
  end SolveLayout
end Terrain
```

constant bound to ensure the question of whether a component may have a given coordinate can be answered in constant time.

Figure 3 illustrates this process with six components added to a grid. The components are identified by a number, and the a component's coordinates are labeled with its number. Claimed coordinates are shown in lightface; taken coordinates, in boldface. The first three components are added by taking empty coordinates. To add component #9 (of size 2), which has the highest affinity for component #4, component #7 renounces one of its claimed coordinates, so that #9 may have it. Similarly, in the sixth step, to add component #10, component #7 renounces a claimed coordinate to allow #9 to renounce one of its claimed coordinates so that #10 may have it. As the figure shows, the ability to renounce claimed coordinates allows components to get closer to their targets than they would if we were to use a pure greedy approach (all coordinates are taken).

3 Limitations and Next Steps

Although software terrain maps meet the desiderata mentioned in the introduction, there are drawbacks. First, basing the size of the methods on the size of the methods' text leads to many methods of size 1. The distribution of the sizes of this library's 3800 methods is an exponential decay curve, which is typical of several systems I measured:



The more size-1 methods there are in a software terrain map, the fewer visual landmarks.

One way to address the problem is to overlay additional landmarks on top of the methods, based, for instance, on the method's control structure. For example, one could add icons like the following, which are analogous to the symbols for schools, campgrounds, etc. found on cartographic maps:

loop

nested loop

switch statement

These three are good candidates in that relatively few methods contain them. Such icons, however, are not useful at the scale of the map in Figure 1 since the individual tiles are too small to contain the icons.

A more serious limitation is that the appearance of the map is based on constraints which change as the software evolves. For instance, when the developer adds a new method, placing this method in the middle of the map would cause a cascade of claim renouncing, which would cause many methods to change shape. The result could be very disorienting. One approach to the problem is to keep all new methods off to the side, ignoring the new methods' affinities, until the developer (or perhaps the whole team) is ready for a large, disorienting map change. This solution does not apply to methods already in the map that are gaining new code. These growing methods would cause similar cascading changes. Experimenting with how visually disorienting users find these cascades is future work. Note that method deletions can be handled by leaving holes. The algorithm described here already generates a few holes (visible both in Figures 1 and 2), which I have allowed as another form of visual landmark.

I have implemented a prototype version of software terrain maps and have integrated it into Microsoft's Visual Studio development environment. The next step is a formal user study to evaluate how well these maps keep users oriented.

4 References

[1] R. DeLine, A. Khella, M. Czerwinski and G. Roberson, "Towards understanding programs through wear-based filtering," *Proc. Symp. on Software Visualization*, 2005.

[2] Furnas, G. "Generalized fisheye views," CHI '86.

[3] Golomb, S. *Polyominoes*, Princeton University Press, 1952.

[4] Storey, M.-A., "SHriMP views: an interactive environment for exploring multiple hierarchical views of a Java program," ICSE 2001 Workshop on Software Visualization, Toronto, Ontario, Canada, May 12, 2001.

From the Concrete to the Abstract: Visual Representations of Program Execution

Steven P. Reiss and Guy Eddon
Department of Computer Science
Brown University
Providence, RI 02912-1910
401-863-7641, FAX: 401-863-7657
{spr,geddon}@cs.brown.edu

Abstract

Programmers have always been curious about what their programs are doing, especially when the behavior is not what they are expecting. Since program execution is intricate and involved, visualization has long been used to provided the programmer with appropriate insights into program execution. This paper looks at the evolution of visual representations of program execution, showing how they have moved from concrete representations of relatively small programs to abstract representations of larger systems. Based on this, we describe the challenges implicit in future execution visualizations and methodologies that can meet these challenges.

1. Introduction

A visual representation of program execution is a graphical display that provides information about what a program is doing *as the program does it*. Visualization is used to make the abstract notion of a computer executing a program concrete in the mind of the programmer. The concurrency of the visualization with the execution lets the programmer correlate real time events, e.g., inputs, button presses, error messages, or unexpected delays with the visualization, making the visualization more useful and meaningful.

Visual representations of program execution have several uses. First, they have traditionally been used for program understanding as can be seen from their use in most algorithm animation systems [7,18]. Second, in various forms they have been integrated into debuggers and used for debugging. Finally, they have often been used as a means of doing performance analysis, visually highlighting program bottlenecks.

What makes a good visual representation depends on the particular application one has in mind. A good representation has to provide the programmer with the data relevant to the task at hand, be it understanding, debugging, or performance analysis, within the limits imposed by the display and the time constraints imposed by concurrency. Since the particular data are often not known in advance, the visualization typically needs to present as much potentially relevant information as possible, and present it in a way so that important or unusual properties stand out visually either directly or through appropriate visual patterns.

This paper is an attempt to describe what is needed to do useful visual representations of the execution of today's software, with an emphasis on understanding. We do this by looking at the different representations we and others have used in the past to learn when and how they are effective and what lessons we can draw from them. We show that the representations have been slowly migrating from the concrete to the abstract. Based on this and the needs of modern systems, we propose the use of programmer-defined abstractions as the basis for a new execution visualization framework.

2. Concrete Representations

The earliest computer-based visualizations showed the actual code as it was executed. These visualizations typically highlighted statements or lines of code as the program was executing that line. These visualization were sometimes combined with other feedback information, for example execution totals or past history.

Many of the early programming environments featured some form of dynamic visualization of the source program. For example, our PECAN environment from the early 1980's outlined the current source statement with a box [8]. This can be seen in the window at the upper right of the display shown in Figure 1. If the program was executing continually, the box kept moving around; if the program was single stepped, the box changed and the program halted with each instruction. Other dynamically updated execution views provided by PECAN included a flowchart view of the program (in the window on the lower right of the figure) and a view of the stack and the values of variables on it seen at the lower left of the figure.

PECAN was followed by algorithm animation systems such as Balsa [2,3], Tango [19], and others that included a view of the source code to highlight what the program was doing. These systems all worked because the programs under consideration were relatively small and execution time was not a primary concern.

After PECAN we tried two different approaches to handling more realistic programs. First, the GARDEN system attempted to do it using abstraction [9,11]. GARDEN was a programming system that let the user define, integrate, and use new visual languages. Each language had a graphical syntax and an execution semantics defined in terms of other languages or GARDEN primitives. Programs were represented by objects that could be

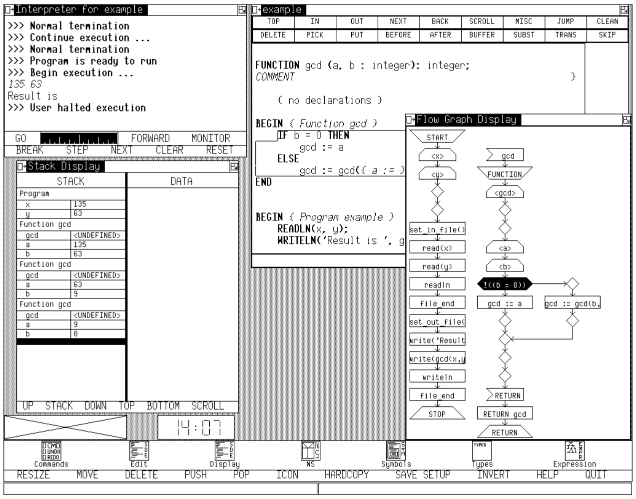


FIGURE 1. The PECAN environment run time visualization.

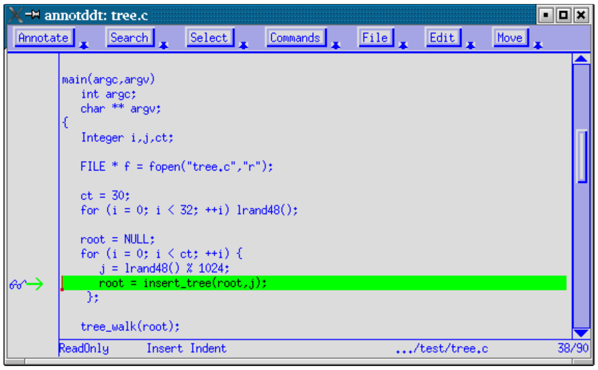


FIGURE 3. FIELD visualization showing source highlighting.

executed directly. GARDEN provided the hooks to automatically highlight execution within the visual displays of a program. Programs were typically constructed using different languages at different levels of abstraction. Since only one level of abstraction was typically displayed in a single window and the user could control the displays, the abstraction level of the visualization was effectively under the control of the user. Figure 2 shows two examples of GARDEN program visualizations, the first a flow chart view and the second a finite state automaton.

Our second approach was in the FIELD system. Here we attempted to provide visualization of full-sized C (and later Pascal, Object Pascal, and C++) systems [12,13]. While most of its visualizations were somewhat abstract (and are covered in the next section), it did source level views that updated whenever the debugger stopped execution. Moreover, it support automatic single stepping so that the user could view the program execution in the editor. This is shown in Figure 3. FIELD offered two types of source highlighting: either the text itself could be highlighted or an appropriate annotation (in this case the green arrow), would move around on the display. The editor would automatically follow execution by changing its focus or file.

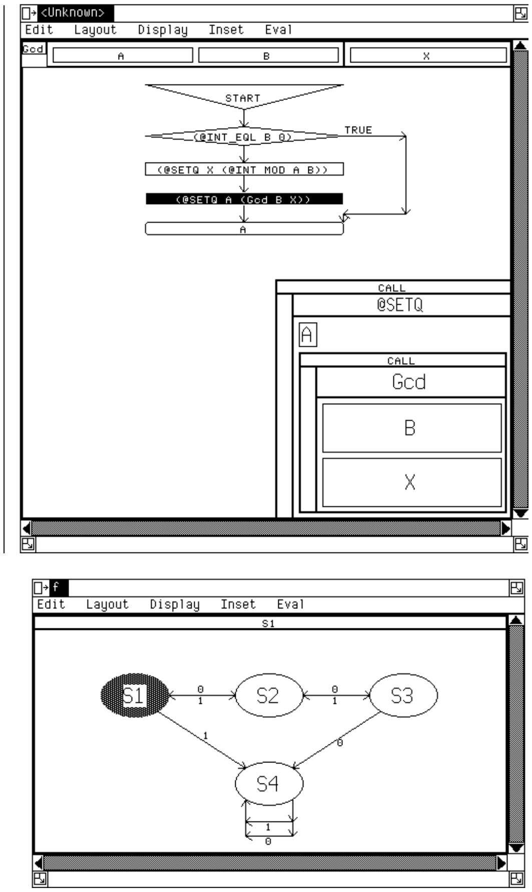


FIGURE 2. Visualization of GARDEN visual programs in action.

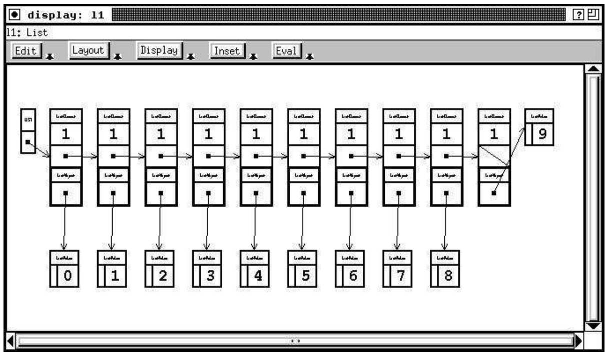


FIGURE 4. FIELD data structure display.

In addition to visualizing the source, FIELD provided visualizations of user data structures that were updated dynamically as the program executed as seen in Figure 4.

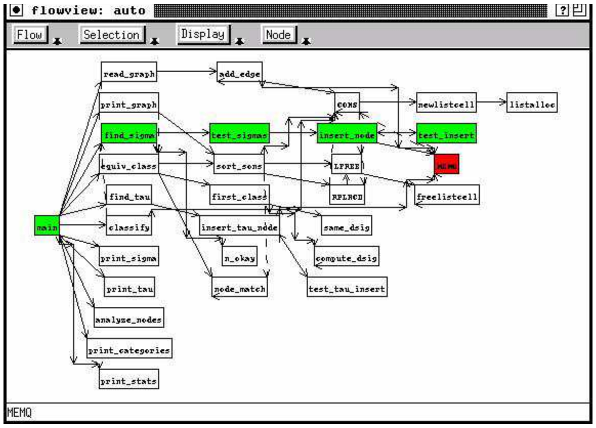


FIGURE 5. FIELD call graph visualization; highlighting shows what is currently executing.

The user was given control over when to update the structure to keep performance reasonable. This is similar to the displays provided by other tools [1,6] and later commercial environments from SGI and Sun. FIELD also let the user customize the data structure displays [10].

These very concrete visualizations of program execution are somewhat helpful, but found limited acceptance and practicality. Lines of code are executed much too rapidly to provide practical views of systems running at or near their normal speed. A program today can easily execute a million lines a second -- far more than can be viewed or even displayed in a meaningful way. Execution speed, when limited to that required to update the display for each line is just too slow for anything other than demonstrations or attempting to understand small programs or algorithms. What was needed was a way of viewing programs that run at closer to their normal speed.

The data structure views had other problems. First, obtaining the information needed to visualize an arbitrary data structure was costly and slowed the program down so much that the various tools updated only when the program was stopped at a breakpoint. Second, real world data structures are too complex to display in a meaningful way without significant user input.

3. Semi-Abstract Representations

Since source lines were too fine a representation to show dynamic execution, visualizations soon moved to more abstract forms. The idea here is to take a higher level view of the program and then to show the execution dynamically in terms of that view.

One obvious high-level view is that of call graphs. The FIELD environment, for example, was able to extract and display the call graph of the system in question. Execution was then shown in the graph by coloring the node currently executing and, optionally, by coloring active nodes (those on the stack) a different color. An example with the current node in red and the stack in green can be seen in Figure 5.

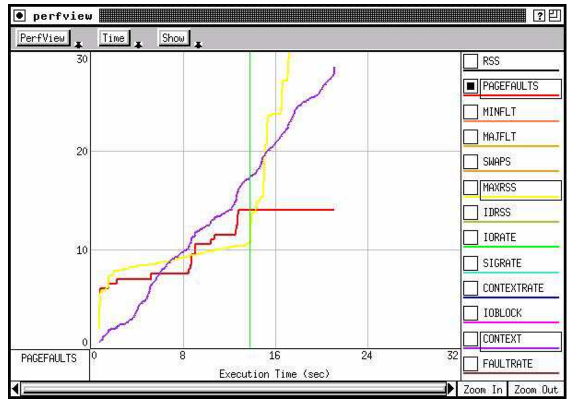


FIGURE 6. Dynamic performance visualization in FIELD.

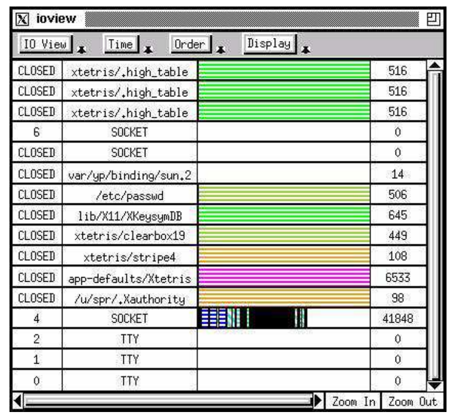


FIGURE 7. IO viewer showing file activity during execution.

To handle large programs, FIELD allowed abstraction within the call graph. A node on the display could represent a single function, a file, a directory, a directory hierarchy, or, for object-oriented systems, a set of methods with the same name in multiple classes. A related view provided by FIELD showed the currently executing method via highlighting in the class hierarchy browser, a predecessor of today's UML class diagrams.

FIELD also provided visualizations that concentrated on performance and on particular behaviors. The performance view, shown in Figure 6, showed the resources that the program was using as it ran. This is closely related but more detailed than the type of views provided by operating system based visualizations such as IBM's PV [5] the visualizations that accompany MPI, or the visualizations incorporated in Sun's programming environment. Information about files and file usage during execution was shown in the file viewer seen in Figure 7. Information about memory was shown in the heap viewer seen in Figure 8. Both the file and memory views updated dynamically as the program ran.

These visualizations were more successful and useful than the earlier direct representations. The call graph and

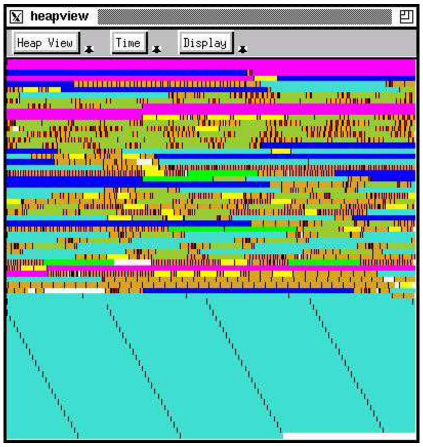


FIGURE 8. Heap visualizer showing memory utilization during execution.

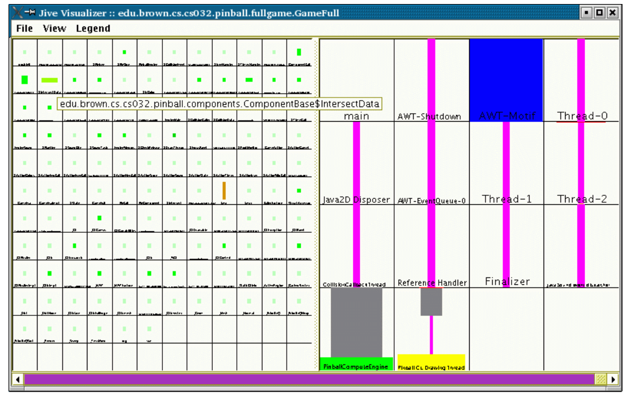


FIGURE 9. JIVE visualization. Class usage is shown on the left; thread usage on the right.

class hierarchy views were typically used by students while working on their class projects. They were not used extensively for larger systems because they did slow the execution significantly, albeit a lot less than highlighting source lines. The specific visualizations for I/O and memory found wider usage, since they could be used with minimal overhead on arbitrary systems. These visualizations were limited however by the limited domains and their lack of history. They were each aimed at specific problems such as identifying files left open or finding memory leaks, and did not extend to more general cases. While some of the motivations for a programmer to use dynamic visualization dealt with these specific problems, many others did so only partially or not at all.

4. Abstract Representations

The heap and file visualizations were successful for real systems because they allowed the program to run at or near full speed while still providing useful information. The main problem with them was that the information was quite limited in that it only touched on one particular domain and thus only helped with understanding or debugging of problems in that domain.

The reason that these views succeed was because they provided what is essentially an abstraction of the program execution. For example, the heap view built its model of memory by only looking at calls to the memory management routines; the IO visualizer did the same by looking only at file open, close, read and write routines. While these abstractions were close to the actual workings of the system, it is possible to use other abstractions to get more complete or more detailed visualizations while still maintaining program performance.

We have been working on such abstractions. Our first system along these lines, JIVE, combines several abstractions into one visualization [15]. One of these abstractions provides a view of execution in terms of classes or packages while the other provides an abstraction of thread behavior. Figure 9 show these two on the left and right respectively. JIVE runs with a slowdown factor of two.

Both of these views model the program behavior over time. The class model breaks up execution into intervals of about 10 milliseconds each. For each interval it keeps track for each class of the number of calls to methods of that class, the number of allocations done by methods of the class, the number of allocations of the class, and the number of synchronizations on objects of the class. The display then shows, for the current interval, the number of calls as the height of the bar, the number of allocations done as the width of the bar, the number of allocations of the class using the hue of the bar, and the number of synchronization as the saturation of the bar. The user can also view totals through the current interval rather than just the values of the interval and can use the scroll bar at the bottom to go back and forth in time.

The thread model on the right views each thread as being in one of eight abstract states: starting, running, running synchronized, blocking, doing I/O, sleeping, waiting, or dead. It tracks the state of each thread over time, maintaining the set of state changes and when they occur. This information can be displayed as on the right of Figure 9 as bars showing the percent of time each thread spends in each state during the current interval (or the totals up through the interval), or it can be displayed as a time graph as seen in Figure 10.

A third model of program dynamic program behavior is seen in the color of the scroll bar in the JIVE visualizations. JIVE uses the information about class and thread usage to try to match the programmer's intuition as to the phases of their program. It uses statistical methods to determine whether the current interval represents a continuation of the existing phase, a reinstatement of a previous phase, or a new phase. It then uses color to display the information about phase changes in the bottom of the window [16].

A second visualizer, JOVE, maintains a more complex model of program behavior [17]. It again looks at the program in terms of small intervals. For each interval it keeps track of how many times each basic block is exe-

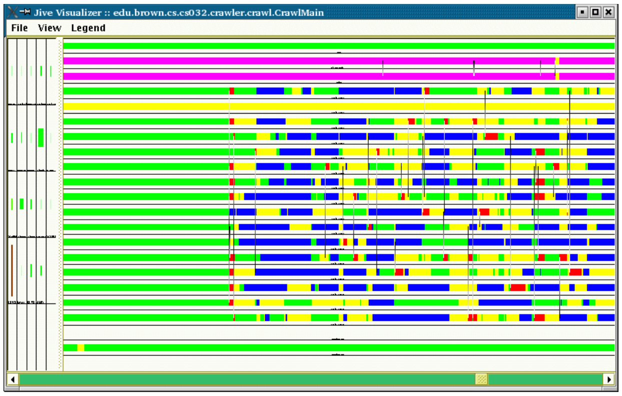


FIGURE 10. JIVE visualization showing thread states along a time line.

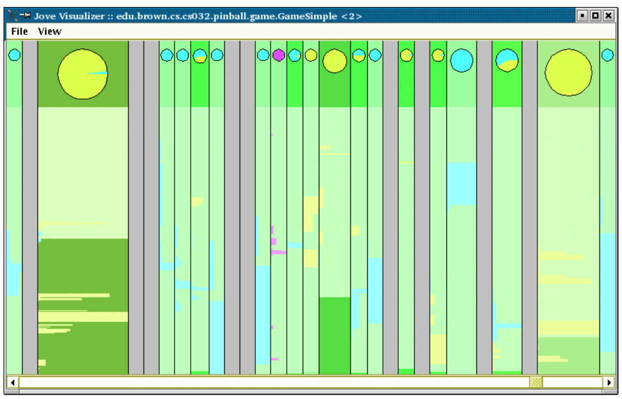


FIGURE 11. JOVE display showing thread usage at the basic block level.

cuted by each thread. The summary information is then kept over the history of the run and is used to produce displays such as seen in Figure 11. Here each vertical region represents a class. The pie chart at the top of the bar is used to show how much time each thread spent in that particular class during the interval. The height of the darker background color for the region indicates the number of allocations done. The lines within the region show information about the various basic blocks. The color of these lines indicate the thread or threads executing those blocks; the width of the line indicates the number of times the block was executed. JOVE slows the program down by at most a factor of four.

The abstract views of JIVE and JOVE are useful for providing the programmer with overview information describing what the program is doing. We have used them for debugging, understanding, and performance analysis. For the latter, they provide useful information about where execution time is spent in the program, either at a high level in JIVE or at a detailed level in JOVE. The high level view was used, for example, to determine that the 3D graphics and gravity computations of a pinball program only used about one third of the available execution time each, and hence were fast enough. The detailed view provided insights into which collision computations were the slowest.

The thread visualizations of JIVE were the most appropriate for debugging and understanding. They readily showed such events as a thread sleeping rather than waiting (and thus blocking other threads) and a thread that was blocking other threads while waiting for I/O. For a multithreaded web crawler, they showed how the threads were divided between waiting for web pages and processing the pages. They also showed the locks that occurred due to synchronization in Sun's HTML parser.

These views however suffer much of the same limits as the I/O and memory visualizations of FIELD. They address specific program aspects (albeit more general ones), and are limited to addressing issues directly related to those aspects. They provide general information about the program execution rather than information that is specific to the particular application or the coding abstractions.

5. Programmer-Defined Representations

The challenge for dynamic program visualizations is to provide information that is meaningful for understanding specific but not yet defined questions while running the application at or near full speed.

Our experiences show that the visualizations that have been most widely used and appreciated for production programs are those that provide a visual model of some aspect of the execution and dynamically update that model as the program runs. These include the memory and I/O visualizers of FIELD and the thread visualizers of JIVE. These systems worked because the model they provide is directly relevant to both the program and to particular problems that are of interest to the programmer. While it is difficult to get a gestalt of the memory behavior of a program from a typical debugger or print statements, the memory visualizer provides such a view at a glance. Through visual patterns it quickly shows memory leaks, abnormally large or unusual allocations, and memory fragmentation. The thread visualizations do similar things related to problems relevant to thread behavior and interaction.

If such views are going to be extended to make dynamic visualization more useful in general, they will have to be based on models that address the issues that programmers want to understand or debug about their particular systems. These models will need to reflect how programmers view their systems, dynamically update these models as the program runs, and then provide visualizations of these models that convey the necessary information.

Such models can be program or language specific. For example, consider a multithreaded web crawler. Each thread repeatedly is assigned a page. It reads that page, parses it, computes summary information, and then stores data about the page based on the parse. Programmers might want to see what each thread is doing in terms of this model. They want to differentiate parsing the page from computing the summary information; they want to know when it is waiting for the web versus waiting to write information to disk. Essentially, they have a model of thread behavior and want to see a visual display of that model.

As a simple example of language-based models, consider iterators in Java. Suppose one wants to track all the currently active iterators in a program, seeing which are currently active, and ensuring each is used correctly, e.g. that *hasNext* is called before *next*. A dynamic visualization could provide a display that showed each active iterator as a box colored by its current state (unused, *hasNext* called, *next* called, *next* called without *hasNext*), with positional information relating the iterators to the source or to particular threads. From such a display the programmer would see what is going on and potential errors would stick out.

These and other visualizations can be provided by letting programmers define the appropriate models for their programs and then providing suitable visualizations. To be practical, the models must be easy to specify and understand, quick to do, and reusable. The set of visualizations provided must be flexible and easily adaptable to the different models. The challenges here involve finding the right framework for defining the models, defining an appropriately broad set of visualizations, providing support to automate or simplify associating the model with the visualization, ensuring that the models can be derived from program execution with small overhead, and doing all of this in a visual framework.

These challenges can be met. Automata over parameterized program events combined with suitable data structures can be used to model the above situations with minimal overhead and with a visual language front end. Multiple visualizations can be defined and associated with models using appropriate heuristics as in [14]. Systems such as Aspect/J show that program events can be detected efficiently [4].

6. Conclusions

Visualizations of program execution have evolved from concrete representations of the source code that were slow and only practical for simple programs, to abstract representations that, while they don't show everything, show detailed information about some aspect of the execution and that work for real systems. To extend the utility of such dynamic visualizations, one needs to look at maintaining and visualizing new abstractions as the program runs. We propose a model whereby programmers can easily define such abstractions that are relevant to their particular understanding or debugging tasks and then have appropriate visualizations generated from these abstractions. This is the next phase of dynamic program visualization.

Acknowledgements. This work was done with support from the National Science Foundation through grants CCR021897 and ACI9982266.

7. References

1. David B. Baskerville, "Graphic presentation of data structures in the DBX debugger," UC Berkeley UCB/CSD 86/260 (1985).

2. Marc H. Brown and Steven P. Reiss, "Debugging in the BALSA-PECAN integrated environment," ACM SIGPLAN-SIGSOFT Symposium on Debugging (1983).

3. Marc H. Brown and Robert Sedgewick, "A system for algorithm animation," *Computer Graphics* Vol. **18**(3) pp. 177-186 (July 1984).

4. G. Kiczales, E. Hilsdale, J. Hugunin, M. Kersten, J. Palm, and W. Griswold, "An Overview of AspectJ," in *European Conference on Object-Oriented Programming*, (2001).

5. Doug Kimelman, Bryan Rosenburg, and Tova Roth, "Visualization of dynamics in real world software systems," pp. 293-314 in *Software Visualization*: *Programming as a Multimedia Experience*, ed. John Stasko, John Domingue, Marc H. Brown, and Blaine A. Price,MIT Press (1998).

6. Brad A. Myers, "Displaying data structures for interactive debugging," Xerox csl-80-7 (June 1980).

7. B. A. Price, I. S. Small, and R. M. Baecker, "A taxonomy of software visualization," *Journal of Visual Languages* Vol. **4**(3) pp. 211-266 (Dec. 1993).

8. Steven P. Reiss, "PECAN: program development systems that support multiple views," *IEEE Trans*. *Soft*. *Eng*. Vol. **SE-11** pp. 276-284 (March 1985).

9. Steven P. Reiss, Eric J. Golin, and Robert V. Rubin, "Prototyping visual languages with the GARDEN system," *Proc*. *IEEE Symp*. *on Visual Languages*, (June 1986).

10. Steven P. Reiss and Joseph N. Pato, "Displaying program and data structures," *Proc*. *20th Hawaii Intl*. *Conf*. *System Sciences*, (January 1987).

11. Steven P. Reiss, "Working in the Garden environment for conceptual programming," *IEEE Software* Vol. **4**(6) pp. 16-27 (November 1987).

12. Steven P. Reiss, "Interacting with the FIELD environment," *Software Practice and Experience* Vol. **20**(S1) pp. 89-115 (June 1990).

13. Steven P. Reiss, *FIELD*: *A Friendly Integrated Environment for Learning and Development*, Kluwer (1994).

14. Steven P. Reiss, "A visual query language for software visualization," *IEEE 2002 Symposium on Human Centric Computing Languages and Environments*, pp. 80-82 (September 2002).

15. Steven P. Reiss, "JIVE: visualizing Java in action," *Proc*. *ICSE 2003*, pp. 820-821 (May 2003).

16. Steven P. Reiss, "Dynamic detection and visualization of software phases," *Proc*. *Third International Workshop on Dynamic Analysis*, (May 2005).

17. Steven P. Reiss and Manos Renieris, "JOVE: Java as it happens," *Proc*. *SoftVis '05*, (May 2005).

18. John Stasko, John Domingue, Marc H. Brown, and Blaine A. Price, *Software Visualization*: *Programming as a Multimedia Experience*, MIT Press (1998).

19. John T. Stasko, "TANGO: a framework and system for algorithm animation," *IEEE Computer* Vol. **23**(9) pp. 27-39 (September 1990).

Design Aspects of the Redwood Programming Environment

Brian T. Westphal, Frederick C. Harris, Jr., Sergiu M. Dascalu
Department of Computer Science and Engineering
University of Nevada, Reno
Reno, NV, 89557 USA
{westphal, fredh, dascalus}@cse.unr.edu

Abstract *— Redwood is a development environment that supports drag-and-drop manipulation of programming constructs and visual representation of program structure. Redwood's architecture and functionality are based on the concept of snippet, defined loosely as a program component that encapsulates both a coding solution and its visual presentation. In addition, snippets support creation of unrestricted code libraries, thus fostering open-source development. This paper presents the motivation for Redwood, briefly overviews its functionality and mode of operation, and then focuses on the concepts underlying its design. Two essential parts of this design are the Snippet Display syntax (SDS) and the Snipplet Language (SL), created by the authors and presented in the paper. Implementation details, examples of use, and several directions of future work are also included in the paper.*

Index Terms *— Development environment, Redwood, snippet, visual programming.*

1 INTRODUCTION

The Redwood programming environment, whose main design elements are presented in this paper, is a project initiated in Spring 2003 at the University of Nevada, Reno. At that time, those involved in this project set forth a number of objectives for the environment, among them enhanced support for hierarchical program design, visualization and direct (via drag-and-drop) manipulation of programming constructs and components, algorithmic independence, inclusion of multiple programming languages, and open source software development [1, 2, 3].

An operational version, Redwood Beta 1, was made available in early 2004. Redwood Beta 1 demonstrated many of the key technologies necessary to implement a usable drag-and-drop programming environment. That release also led to formulations of new ideas and realizations that certain aspects have to be modified in order to create a truly functional development product. Beta 1 introduced the concepts of snippets, design trees, and disclosure dots [1]. Snippets provide a means by which generic programming constructs could be described. Design trees describe, in part, the relationships between the program components that make up a program. Lastly, disclosure dots, inspired from disclosure triangles in Mac OS X [4], work with snippets and design trees to allow visualization of source code at various levels of abstraction.

One of the most important lessons learned in developing the first release of Redwood, is that screen real estate is a precious resource for programmers [5, 6]. For Beta 2, the entire snippets engine had to be rethought and rewritten to better address this. Another lesson that became apparent while developing complicated programs using Redwood Beta 1 was that static templates are not sufficient for describing generic programming constructs. With the Beta 2 release, made publicly available in May 2005, one can create very powerful, dynamic templates.

This paper presents recent results related to Redwood's latest version. Notably, although the authors remained truthful to the key concepts that define Redwood's development (snippets, design tree, disclosure dots, and drag-and-drop programming) they have radically re-designed the environment, which in terms of interface is currently refined to its most essential and elegant version to date. In terms of snippet manipulation and presentation, the new version is also significantly more powerful. The experience gained in undertaking Redwood's recent overhaul is reported here, through the presentation of the environment's new "look", details on the two "internal notations" created to build the latest version of Redwood (the Syntax Display Language and the Snipplet Language), and examples of use.

The remainder of this paper is organized as follows: Section 2 describes the motivation for creating Redwood and presents an overview of the environment, Section 3 outlines the principal architectural solutions used to build the environment, Section 4 provides details about the Snippet Display Syntax, Section 5 focuses on the Snipplet Language, Section 6 uses the sigma summation example to illustrate how programs are developed in Redwood, Section 7 briefly compares Redwood with related projects, and Section 8 concludes the paper with pointers to future work and a summary of the environment's significance.

2 REDWOOD: RATIONALE AND OVERVIEW

From its very beginning, the Redwood programming environment (whose main browser is shown in Figure 1) has been intended as a tool that provides support for the large community of open source software developers. Another main idea behind Redwood's design has been to provide a user interface that is easy to learn, easy to use, quick to operate, and highly reliable. These two major objectives of Redwood's design have both been aimed at increasing the developers' efficiency as well as at enhancing their satisfaction when creating programs with this environment.

Redwood's distinguishing characteristics, particularly its support for graphical representation and direct manipulation of various program constructs and components, allows software creation in a visual workspace that is more intuitive and effective than that of a regular plain text code editor. In Redwood, developers can easily select program components (snippets) by clicking on their names in a tools panel (snippet chooser) and then drag-and-drop them to that place in the program's layout (program's structure) deemed to be the best for satisfying the program's requirements. Notably, the collection of snippets used for building programs in Redwood could grow constantly, as Redwood's library could be updated regularly via the Internet. Shown in Figure 1 are the snippet chooser plug-in panel (on the left-hand side of Redwood's main browser) and the editing panel (on the right-hand side of the browser).

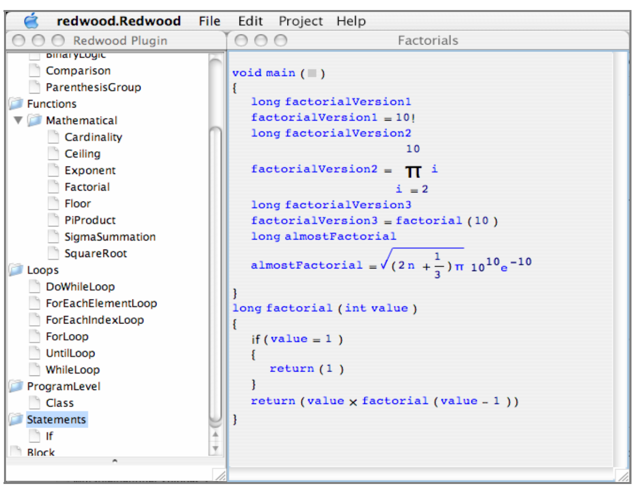


Fig. 1. Redwood's main browser

These are the core tools in Redwood for, respectively, selecting and organizing snippets. Note that the program shown in the editing panel of Figure 1 is included to illustrate the manipulation of various snippets. A more meaningful program in terms of algorithmic content is presented in Section 6 of the paper.

Writing and manipulating source code is often ineffective if it involves dealing with a large number of abstractions. With open source software (OSS) part of this problem is alleviated, as developers have access to a significant amount of code already written [7, 8, 9]. Therefore, they can avoid rewriting large portions of code and focus instead on the innovative aspects of their software. Unfortunately, software available as open source has its own drawbacks, for example it can be difficult to locate, poorly documented, not sufficiently supported, and not stable enough [8, 10, 11]. The Redwood environment attempts to solve some of these problems, as its architecture and functionality is intended not only for efficient program construction but also for effective manipulation of snippets, which are well-suited entities for supporting open source software development.

The snippets technology of Redwood, detailed later in the paper, is designed for promoting the reusability of code. Snippets give developers the power to encapsulate ideas, not just classes or functions, and to visualize program syntax in meaningful ways. For example, with a snippet one can use images or drawings to represent design concepts for which one might implement the code later. In addition, a future release of Redwood will be configurable so that software documentation can be enforced. These features of Redwood could help solve problems with open source development and programming in general. Developers do not have to rewrite time and again essentially the same code, as it could be made available through the environment's interface, both locally or remotely. In many software projects, customization of existing snippets could represent the only programming effort needed.

3 INTERNAL DESIGN SUPPORT

Snippets, the key elements of Redwood's design philosophy, can be described loosely as software components that encapsulate a coding solution (program logic) and its associated graphical representation (data for on-screen visualization). Snippets have been the subject of a previous publication, to which we refer the interested reader [2]. We mention only that a snippet, which can be as simple as an assignment statement or as complex as a very intricate algorithm, has an internal structure that consists of two parts: a display section and a template section.

A snippet's display and template sections are described using the specially created Snippet Display Syntax (SDS) and Snipplet Language (SL, or simply Snipplet), respectively. The display section of a snippet defines the portions of the snippet that are visible to the user and the template section specifies the mapping from the visualized snippet to the programming language output (generated code). As detailed later in the paper, display representations are currently described using static XML while template sections are described using dynamic Snipplet scripts.

The SDS was designed to be a general-purpose interface description that is both easy to write and simple to parse. The SDS came about as part of a supporting technology designed by the authors for Redwood, the interface builder (now encapsulated as the com.bleugris.xml package [12]). The SDS is a convenient way to build interfaces without writing Java code, which can be tedious. This syntax allows one to place into the interface description any Java AWT or Swing components, including custom components, snippets, and snippet editors. The interface builder allows one to load interfaces dynamically from files.

The SL was devised for generating source code. As such, its design needed to revolve around parsing and string handling. In addition, scripts needed to have dynamic access to the elements of the design tree. The language was built primarily with two other scripting languages in mind, PERL [13] and JavaScript [14]. These languages are commonly used for generating HTML and JavaScript code for web pages. Still, they are not fully geared towards source code generation. In particular, code in both languages tends to become unstructured quickly, if not carefully managed. For source code generation, dealing with "languages inside of languages", this was something that had to be avoided. Like PERL and JavaScript, the SL makes use of variant types and has several built-in functions specifically designed for parsing. However, unlike PERL and JavaScript, SL's set is kept to a minimum, removing insecure features such as file I/O. The SDS and the SL are detailed, respectively, in the next two sections of the paper.

4 SNIPPET DISPLAY SYNTAX

A snippet is defined, using XML, in two sections. The first one, the display section, is immediately apparent to the user of the snippet (through visualization). The second one, the template section, is only noticeable when the user builds a program containing a snippet. This section discusses SDS, the syntax used for defining the display section of a snippet.

Within SDS, snippet display tags contain table-based layout of snippet editors. Using a table-based layout allows one to create non-linear or two-dimensional layouts. The structure of the table layouts is similar to HTML-style tables. That is, tables are organized into rows, and rows are organized into columns – they are "row major". Each cell can span one or more rows and columns and can be given width and height attributes. In addition, each cell can be given alignment parameters in both the horizontal and vertical directions.

Snippet editors are the foundation of snippet displays. Each snippet must contain at least one snippet editor. Snippet editors are special Java Swing components that extend the SnippetEditor class. Several snippet editors are included in Redwood. Each has various parameters that may be set. For example, the LineEditor snippet editor is commonly used to display a single line of text. The Editable parameter can be set to false so that the editor is used as a display only, and not as an input. The Text parameter is used to set the message of the editor. The complete description of SDL has been included in an internal report at [3] and is available upon request.

5 SNIPPLET LANGUAGE

The Snipplet Language (SL, or just Snipplet) was created specifically to support source code generation. For inspiration, ideas from two languages commonly used for code generation, PERL and JavaScript, were used. The PERL or Practical Extraction and Report Language [13] is often used for CGI programming [15]. CGI allows one to dynamically create web pages, through generation of code such as HTML, JavaScript, and CSS. The JavaScript language, a derivative of Java, is often used to generate HTML and JavaScript code on the client-side [14]. Both languages have features and syntax that lend nicely to source code generation. In addition, several original ideas have been incorporated into the Snipplet language design.

Similar to JavaScript, variables in Snipplet are of variant type, meaning they can switch between types. The standard data types include integers, real numbers, strings, arrays (including multi-dimensional), and associative arrays (or hash tables). In addition to these, it is also possible to access snippets, giving scripts direct access to the design trees of programs.

While examining the features that a source code generation language should have, several significant factors come to mind. Most importantly, the syntax should be direct and simple, able to "stay out of the developer's way." Because one is working with two levels of syntax – the syntax for the language being used (Snipplet) and the syntax for the language being generated – it should be easy to distinguish between the two levels. At the same time, the language should be powerful enough and flexible enough to simplify the often-complex demands involved in source code generation.

Another important factor in designing a language for generating code is to provide a tendency for self-organizing syntax. That is, Snipplet code should not become confusing to examine due to lack of organizational formatting or overly compressed syntax. PERL programs often suffer from use of rather esoteric functions. Even though one can look up the meanings of various shortcut functions, inclusion of these in a language designed specifically for source code generation would be a mistake. Examples include the familiar "s///" function in PERL, which is a function for replacing substrings. Although compressed syntax can be convenient for the programmer, more straightforward naming conventions let programmers less experienced with the language or program interpret and modify code more easily. In fact, in Snipplet's case, even though the language is quite powerful, the grammar and the list of functions that a developer needs to know to work with it are relatively short.

The grammar for the Snipplet Language is shown, broken down into levels, in Figures 2 through 7. Level 0, shown in Figure 2, is the start symbol for the grammar. As one reaches higher levels in the grammar, productions are more specific.

```
1  template ::= block WS
```

Fig. 2. SL Grammar: Level 0, the start symbol

```
2  block ::= statement*
3  WS ::= ('\s+' | '//.*' | '/\*([\u0000-\u0029\u002B-
   \u9999]|\*[\u0000-\u002E\u0030-\u9999])*\*/')*
```

Fig. 3. SL Grammar: Level 1, basic template components

```
4  statement ::= WS (break | declaration | return |
   expression) WS ';' | WS (functionDeclaration | if |
   loop) WS
```

Fig. 4. SL Grammar: Level 2, statement types

```
5  break ::= 'break'
6  declaration ::= 'var' SP identifier (WS '=' WS
   (arrayInitializer | expression))?
7  expression ::= binaryArithmeticLevel5

8  //Sub-expressions
9  binaryArithmeticLevel5 ::= binaryArithmeticLevel4
   binaryArithmeticLevel5a?
10 binaryArithmeticLevel5a ::= WS '\|\|' WS
   binaryArithmeticLevel4 binaryArithmeticLevel5a?
11 binaryArithmeticLevel4 ::= binaryArithmeticLevel3
   binaryArithmeticLevel4a?
12 binaryArithmeticLevel4a ::= WS '&&' WS
   binaryArithmeticLevel3 binaryArithmeticLevel4a?
13 binaryArithmeticLevel3 ::= binaryArithmeticLevel2
   binaryArithmeticLevel3a?
14 binaryArithmeticLevel3a ::= WS '!=|<|<=|==|>=|>' WS
   binaryArithmeticLevel2 binaryArithmeticLevel3a?
15 binaryArithmeticLevel2 ::= binaryArithmeticLevel1
   binaryArithmeticLevel2a?
16 binaryArithmeticLevel2a ::= WS '\+|\-' WS
   binaryArithmeticLevel1 binaryArithmeticLevel2a?
17 binaryArithmeticLevel1 ::= binaryArithmeticLevel0
   binaryArithmeticLevel1a?
18 binaryArithmeticLevel1a ::= WS '\*|/|%' WS
   binaryArithmeticLevel0 binaryArithmeticLevel1a?
19 binaryArithmeticLevel0 ::= mainExpressionPart
20 //End sub-expressions

21 return ::= 'return' SP expression
22 functionDeclaration ::= 'sub' SP identifier WS '\{' WS
   block WS '\}'
23 if ::= ifPart (WS elseIfPart)* (WS elsePart)?
24 loop ::= dowhileLoop | foreachLoop | forLoop |
        whileLoop
```

Fig. 5. SL Grammar: Level 3, statement specifications

```
25 identifier ::= arrayIdentifier | scalarIdentifier |
   hashIdentifier
26 arrayInitializer ::= '\{' WS (expression (WS ',' WS
   expression)* WS)? '\}'
27 typeCast ::= '\(' WS ('string' | 'real' | 'integer') WS
   '\)'
28 mainExpressionPart ::= (typeCast WS)?  (assignment |
   doubleValue | functionCall | identifier | longValue |
   parentheses | string | unaryArithmetic |
   snippetEditorFunctionCall)
29 ifPart ::= 'if' WS condition WS '\{' WS block WS '\}'
30 elseIfPart ::= 'elseif' WS condition WS '\{' WS block
   WS '\}'
31 elsePart ::= 'else' WS '\{' WS block WS '\}'
32 dowhileLoop ::= 'do' WS '\{' WS block WS '\}' WS
   'while' WS condition
33 foreachLoop ::= 'foreach' SP identifier WS '\(' WS
   expression WS '\)' WS '\{' WS block WS '\}'
34 forLoop ::= 'for' WS '\(' (expression | declaration) WS
   ';' WS expression WS ';' WS expression WS '\)' WS '\{'
   WS block WS '\}'
35 SP ::= ('\s+' | '//.*' | '/\*([\u0000-\u0029\u002B-
   \u9999]|\*[\u0000-\u002E\u0030-\u9999])*\*/')+
36 whileLoop ::= 'while' WS condition WS '\{' WS block WS
   '\}'
```

Fig. 6. SL Grammar: Level 4, primary support for statements

6 WRITING PROGRAMS IN REDWOOD

To use an existing snippet, one may simply drag a snippet, listed by name in the snippet chooser tool, and drop it into the editing space. Once in place, a snippet can be repositioned and manipulated as needed. A newly dropped snippet is called a visualized snippet. These snippets are ready for customization. Not all snippets require customization; some may meet one's needs immediately upon being dropped. However, most snippets will need at least minor customizations. A snippet can be customized in two ways. With some types of snippet editors, editing text and/or manipulating controls will help customize the snippet. For other types of snippet editors, one may drop additional snippets. For example, in a CodeEditor one may drop as many snippets as necessary. In an ExpressionEditor only a single snippet may be dropped.

```
37 arrayIdentifier ::= scalarIdentifier ('\[' expression?
   '\]')+
38 assignment ::= identifier WS ('=' | '\+=' | '\-=' |
   '\*=' | '/=' | '%=') WS expression
39 doubleValue ::= '(\+|\-)?[0-9]+\.[0-9]+'
40 functionCall ::= identifier WS '\(' WS parameters? WS
   '\)'
41 hashIdentifier ::= scalarIdentifier '\{' expression?
        '\}'
42 longValue ::= '(\+|\-)?[0-9]+'
43 parentheses ::= '\(' expression '\)'
44 scalarIdentifier ::= '[\$@A-Za-z_][A-Za-z_0-9]*'
45 snippetEditorFunctionCall ::= '\[#' WS expression (WS
   ':' WS parameters)? WS '\]'
46 string ::= '"(\\"|[\u0000-\u0021\u0023-
   \u9999])*"|\'(\\\'|[\u0000-\u0026\u0028-\u9999])*\''
47 unaryArithmetic ::= '!' WS expression
48 condition ::= '\(' WS expression WS '\)'
```

Fig. 7. SL Grammar: Level 5, secondary support

The Select Build Language option of the Project menu allows a programmer to select the language desired for output. The current Beta 2 release of Redwood supports C, C++, and Java output for all included snippets. The Project menu's Build option runs the Snipplet scripts for the project generating source code output in the desired language. This code is placed in source files as appropriate.

Figures 8 and 9 form the two portions of the Sigma Summation snippet. Figure 8 contains the display tag, which describes the visual elements of the snippet. Figure 9 contains the template tag. The template tag describes how source code is generated based on the customizations made to the snippet. Figure 10 demonstrates the use of the Sigma Summation snippet in Redwood. The code in the figure performs the naive matrix multiplication algorithm [16]. Because of the graphical environment, programming constructs can be described in their natural form. Figure 11 is the equivalent code written in Java. While it is relatively easy for a programmer to decipher either version, Figure 10 displays a more compact depiction.

7 RELATED WORK

In its early stages, Redwood has been inspired primarily by Alice, a visual programming system developed at Carnegie Melon University by the Stage 3 Research Group [17, 18]. Alice proposed the idea of visual, drag-and-drop programming as a means for teaching computer-based problem solving and computer programming to high school and university students. However, Redwood is not designed specifically for use by students. Redwood's scope is larger, as it aims at providing a useful tool for the much larger open source software community, including beginners, intermediate, and experienced programmers. Of course, based on its main features (visual representation and direct manipulation), Redwood fits in the category of visual programming environments [19, 20].

```
<Snippet type="Math.Sigma Summation"
extends="Statement, Expression"><display><table>
 <tr>
  <td width="8"/>
  <td valign="center">
   <table>
    <tr><td halign="center"><ExpressionEditor
     name="haltingvalue">
     <param name="MinimumSize"><Dimension>
      <param name="Size"><int value="10"/><int
       value="10"/></param>
     </Dimension></param>
    </ExpressionEditor></td></tr>
    <tr><td halign="center"><LineEditor>
     <param name="Text"><String
      value="&#x2211;"/></param>
     <param name="FontSize"><int value="24"/></param>
     <param name="Editable"><boolean
      value="false"/></param>
    </LineEditor></td></tr>
    <tr><td halign="center"><table><tr>
     <td><IdentifierEditor name="loopvariable">
      <param name="Text"><String value="i"/></param>
      <param name="MinimumSize"><Dimension>
       <param name="Size"><int value="4"/><int
value="-
        1"/></param>
      </Dimension></param>
     </IdentifierEditor></td>
     <td><LineEditor>
      <param name="Text"><String value=" =
"/></param>
      <param name="Editable"><boolean
       value="false"/></param>
     </LineEditor></td>
     <td>
      <ExpressionEditor name=”initialvalue”>
       <param name=”Snippet”>
        <Snippet type=”Numeric Value”>
         <LineEditor name=”value”>
           <param name=”Text”><String
            value=”0”/></param>
         </LineEditor>
        </Snippet>
       </param>
       <param name=”MinimumSize”><Dimension>
        <param name=”Size”><int value=”10”/><int
         value=”10”/></param>
       </Dimension></param>
      </ExpressionEditor>
     </td>
    </tr></table></td></tr>
   </table>
  </td>
  <td valign=”center”>
   <ExpressionEditor name=”expression”>
    <param name=”MinimumSize”><Dimension>
     <param name=”Size”><int value=”10”/><int
      value=”10”/></param>
    </Dimension></param>
   </ExpressionEditor>
  </td>
  <td width=”4”/>
 </tr>
</table></display>
```

Fig. 8. Snippet display of Sigma Summation

```
<template language="Java">
 sub calculateReturnValue
 {
  var uid = getUID ();

  var expressionReturnValue = [#"expression" :
   "+Expression:return calculateReturnValue ();"];

  var resultType = [#"expression" :
"+Expression:return
   determineResultType ();"];

  var result[] = {
   "
   " + resultType + " Snippet_SigmaSummation_sum_" +
   uid + " = new " + resultType + " ();
   for (RedwoodDouble " + [#"loopvariable"] + " = " +
    [#"initialvalue"] + ";
   " + [#"loopvariable"] + ".lessThanOrEqualTo (" +
   [#"haltingvalue"] + ").booleanValue (); " +
   [#"loopvariable"] + " = " + [#"loopvariable"] +
   ".add (new RedwoodDouble (1)))
    {
     " + expressionReturnValue[0] + "
     Snippet_SigmaSummation_sum_" + uid + " =
      Snippet_SigmaSummation_sum_" +
     uid + ".add (" + expressionReturnValue[1] + ");
    }
    ",
    "Snippet_SigmaSummation_sum_" + uid
   };
   return result;
  }

  var returnValue = calculateReturnValue ();
  return returnValue[0];
 </template></Snippet>
```

Fig. 9. Snippet template of Sigma Summation (Java output)

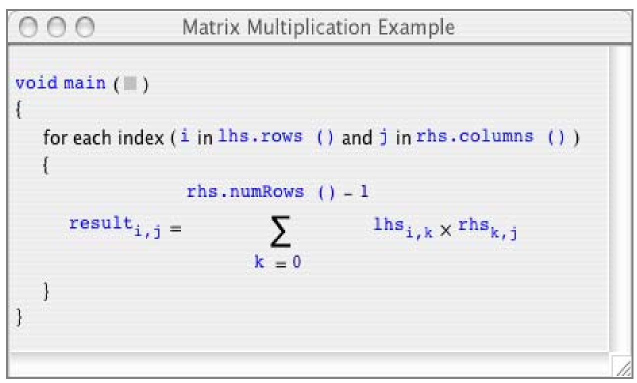


Fig. 10. Matrix Multiplication with Sigma Summation Snippet

```
public static void main (String args[])
{
    for (int i = 0; i < lhs.numRows (); i++)
    {
        for (int j = 0; j < rhs.numColumns (); j++)
        {
            double sum = 0.0;
            for (int k = 0; k < rhs.numRows (); k++)
            {
                sum += lhs.get (i, k) * rhs.get (k, j);
            }
            result.set (i, j, sum);
        }
    }
}
```

Fig. 11. Matrix Multiplication in Java

Examples of environments that have similarities with Redwood include Prograph [21], LabView [22], CODE [23] and several others. Nevertheless Redwood is distinct from them in at least one of the following: execution model, application domain (e.g., LabView is best suited for engineering applications: test, measuring and control [24]), program components used (in this respect, snippets seems to be a concept quite unique to Redwood, at least in the sense we use it), and user interface ("look and feel"), which is clearly Redwood specific. This is not to say that Redwood is better than these languages, as for example the above environments supports parallel programming while Redwood does not at this point in time.

It is fair to say that, first, we are currently concerned with actually enhancing Redwood's present capabilities and plan for the near future a comparative study with other visual environments and, second, most of the above environments are specialized, and good or very good in some respects. We believe that Redwood is also good at what it does (or is characterized by) including support for general program development, "smooth" (that is, streamlined, simple and refined) user interface, support for multiple languages, extensibility, and flexibility.

8 Future Work and Conclusions

In this paper the main design solutions used for creating Redwood were described. We believe that this novel programming environment, based on the key concept of snippets and employing the "drag-and-drop programming" paradigm, offers attractive, efficient and comprehensive support for software development.

Future releases of Redwood will include additional programming support and more usability features. Snippet displays will allow for dynamic content and the environment will be stress-tested for efficiency, reliability, and ease-of-use. Besides general improvements that can be made to the system, we have plans to include a large collection of pre-built snippets in the environments, in addition to the online snippet library. This set will include tools for parallel programming, mathematical representations of programming constructs, and templates for commonly-used structures such as loops.

One of the most useful features of Redwood is that it allows one to "extend" languages. With a plain text programming language, a developer is confined to standards sometimes defined decades before. The developer can add instances of new structures such as classes and functions, but he or she cannot create new types of structures. He or she cannot, for example, create a new type of loop in C++ (such as an until loop), which might be more direct (than a while loop) for solving certain problems. Redwood is about designing and programming software using natural techniques, where the environment and language do not get in the developer's way (on the contrary, it is intended to gracefully and efficiently support him or her). The system creates a supporting environment for a programmer to work and think effectively.

References

[1] Westphal, B. T., Harris, F. C., Jr., and Dascalu, M. S. (2004) "Redwood: A Visual Environment for Software Design and Implementation", in WSEAS Trans. on Computing, 2(3):380-386.

[2] Westphal, B. T., Harris, F. C., Jr., and Dascalu, M. S. (2004) "Snippets: Support for Drag-and-Drop Programming in the Redwood Environment", in Journal of Universal Computer Science (JUCS), 2004, 10(7):859-871.

[3] Westphal, B. T. (2004) "The Redwood Programming Environment", Master's thesis, University of Nevada, Reno, NV, USA.

[4] "Mac OS X Panther Operating System", retrieved March 1, 2005 at http://www.apple.com/macosx/

[5] Burnett, I., Baker, M.J., Bohus, C., Carlson, P., Yang, S., and Van Zee, P. (1995) "Scaling Up Visual Programming Languages", IEEE Computer, vol. 28, no. 3, March, pp. 45-54.

[6] Na, L., Hosking, J, and Grundy, J. (2004) "Integrating a Zoomable User Interfaces Concept into a Visual Langauge Meta-Tool Environment", Proceedings of the 2004 IEEE Symposium on Visual Languages and Human Centric Computing, pp. 38-40.

[7] Wang, H. and Wang, C. (2001) "Open Source Software Adoption: A Status Report", IEEE Software, vol. 18, no.2, pp. 90-95.

[8] Lawton, G. (2002) "Open Source Security: Opportunity or Oxymoron?", IEEE Computer, vol. 35, no. 3, March, pp. 18-21.

[9] van Krogh, S. G., Spaeth, S., and Haefliger, S. (2005) "Knowledge Reuse in Open Source Software: An Exploratory Study of 15 Open Source Projects", Proceedings of the 38th Annual Hawaii International Conference on Systems Sciences (HICSS'05), pp. 1-10.

[10] Ganssle (2000) The Ganssle Group. "Open Source?", retrieved February 20, 2005 at http://www.ganssle.com/articles/opensrc.htm

[11] Fitzgerald, B. (2004) "A Critical Look at Open Source", IEEE Computer, vol. 37, no. 7, July, pp. 92-94.

[12] "com.bleugris.xml v1.07 Programmer's Manual", retrieved March 1, 2005 at http://www.bleugris.com/java/docs/com_bleugris_xml.html

[13] "Perl.com: Documentation", retrieved February 20, 2005 from http://www.perl.com/pub/q/documentation

[14] "Core JavaScript Reference 1.5", retrieved Feb. 2005 from http:// devedge.netscape.com/library/manuals/2000/javascript/1.5/reference

[15] Kew, N. (2000) "CGI Programming FAQ", retrieved February 22, 2005 from http://www.htmlhelp.com/faq/cgifaq.html

[16] Foster, I. (1995) "Matrix Multiplication", retrieved February 28, 2005 from http://www-unix.mcs.anl.gov/dbpp/text/node45.html

[17] Pausch, R. et al. (1995), "A Brief Architectural Overview of Alice, a Rapid Prototyping System for Virtual Reality", IEEE Computer Graphics and Applications, May 1995.

[18] "Alice: Free, Easy, Interactive 3D Graphics for the WWW", retrieved February 28, 2005 from http://www.alice.org/

[19] Levialdi, S. (2001) "Visual Languages: Concepts, Constructs, and Claims", Proceedings of the 23rd IEEE International Conference on Information Technology Interfaces, pp. 29-33.

[20] Burnett, M.M. (2001) "Software Engineering for Visual Programming Languages", Handbook for Soft. Eng. and Knowledge Engineering, vol. 2, pp. 77-92. World Scientific Publishing.

[21] Cox, P.T., Glaser, H., and MacLean, S. (1998) "A Visual Development Ernvironment for Parallel Applications", Proceedings of the 1998 IEEE Symposium on Visual Languages, pp. 144-151.

[22] Wang, J.Z. (2003) "LabView in Engineering Laboratory Courses", Procs. of 33rd IEEE Frontiers in Education Conf., vol. 2, pp. FE13.

[23] Newton, P. and Browne, J.C. (1992) "The CODE 2.0 Graphical Parallel Programming Language", Proceedings of the ACM International Conference on Supercomputing, pp. 167-177.

[24] Josifovska, J. (2003) "The Father of LabView", IEE Review, Virtual Instrumentation, October, pp. 32-35.

Comic Strip Programs: Beyond Graphical Rewrite Rules

Mikael Kindborg and Kevin McGee
Department of Computer and Information Science
Linköping University, S-58183 Linköping, Sweden
{mikki, kevmc}@ida.liu.se

Abstract

Comics and programs both are static representations of something dynamic. While a comic book almost looks and feels like an animated cartoon, the source code of a program seldom resembles the visible runtime behaviour. By using comic strips to represent concurrent events, graphical programs that feature interactive, animated characters and objects can be expressed in a visually direct way, making it easier for both children and adults to create their own computer programs. Compared to graphical rewrite rules, comic strip programs have a potential for increased expressiveness and flexibility, because of how the semiotics of comics can be used to express a wide variety of situations. There are also challenges with using comics as a program representation, for example, comics have no signs for conditionals, concurrency, and generalisations.

1. Introduction

One approach for making programming easier, is to use a program representation that looks similar to and directly maps to the runtime representation [1]. If the source code of a program directly maps to and has a strong similarity to that which is seen on the display when running the program, the gap between the two representations can be bridged, and the need for difficult and error-prone mental transformations could be reduced, thus making programming easier.

A visual representation that is interesting in this respect is comics. Like a program, a comic is a static representation of something dynamic. The medium of comics gives a very direct impression of the action going on in the story. To the comic book reader, the characters in a comic almost look like they are moving and they almost sound as if they are speaking. For programs that consist of interactive graphical objects, like games, graphical simulations, and interactive pictorial stories, the signs used in comics have the potential to describe the behaviour of the objects in a way that strongly resembles the visual result of running the program.

In comics, panels are used as the basic temporal device for communicating dynamics in a static medium [2], [3]. A static representation is straightforward to edit, which is essential to programming, and provides an overview that is independent of the real-time flow of a dynamic representation. Inside panels, comic book artists use a rich vocabulary of contextual signs ("markers") for representing dynamic and abstract features. Examples include motion markers (e.g. speed lines and ghost images), voice balloons, and sound words (onomatopoetic symbols). Importantly, such markers are shown in the immediate visual context of the character or object having the feature represented by the sign. This presentation technique takes advantage of the way human perception can quickly perceive a situation "at a glance", and creates a high degree of visual directness. Additionally, many people are familiar with comics, and even though there are cultural differences, for example between Western European and Japanese comics [3], the sign language of comics is well-known within the comic book reading community. This could mean that people who want to learn to program would feel familiar with the signs used in a comic strip programming language.

One way to use comics for programming of graphical objects, is to have conditional comic strips that represent events in a program [4]-[7]. Such comic strip programs share many characteristics with graphical rewrite rules [1], [8]-[16], also called visual before-after rules. The before-part of a rule consists of a picture representing a part of the world. When the before-part matches the current world state, the world is changed to the state shown by the picture in the after-part of the rule. Such rules look very similar to what actually happens when the program executes. Graphical rewrite rule systems commonly use a grid-based world model, which makes it straightforward to specify the before and after parts of a rule in a spatially clear and non-ambiguous way.

Graphical rewrite rules are an example of an analogical representation, a representation that has a structural similarity to what it represents [1]. In semiotics, such representations are called iconic signs. An icon is a sign that has a similarity or likeness to the object it signifies; it imitates the signified. A symbolic sign, by contrast, is arbitrary and gets its meaning by convention [17].

Analogical representations and iconic signs are usually easy to understand, but not everything can be represented this way. In programming, there are constructs that have no iconic representation. Comics offer a way to integrate symbolic signs into the context of iconic signs, which can help understanding symbolic representations.

This paper presents a way to use comics for event-based visual programming, and discusses how the signs of comics can be applied to programming. Finally, it is suggested that comic strip programs have the potential to go beyond graphical rewrite rules in terms of expressiveness and flexibility, while maintaining a strong visual directness between the program and the runtime display.

2. Comic strip programming

Several prototypes have been developed to test the idea of comic strip programming. Both low-fidelity paper prototypes and very high-fidelity computer prototypes have been studied [4]-[7]. The domain focused by these prototypes have been simple games and interactive stories with graphical objects. In the following, examples of comic strip programs will be given using screenshots from the most recent system being developed, called ComiKit.

In comics, the action sequences that make up the story are communicated using strips of panels, where each panel shows a part of the action. If we introduce the notion of conditional strips, comic strips can be used to describe events in a program. In a conditional comic strip, the first panel is the precondition for the actions in the subsequent panels. Such strips describe non-linear and potentially concurrent events, rather than a sequential flow of actions in a story.

A comic strip program consists of graphical objects that typically depict figurative characters and items. An object can have any graphical representation, though, as it is the programmer who draws or chooses the pictures. Each object can have one or more pictures. For example, a character could have two pictures, one for happy mood and one for sad mood. The pictures represent different states of an object.

Each object may have one or more comic strip events associated with it. The program is run by placing characters on a play area and moving them with the mouse, causing events to trigger. Events are continuously monitored and executed by the runtime engine (the interpreter).

Figure 1 shows an example of event strips created by two girls in fifth grade, during a field study in a school. Figure 2 shows what it looks like when the game is played. The preconditions used in Figure 1 are picture-matching and touching another object. The action used is picture-changing. Picture matching works by checking if an object has the picture shown in the precondition. If, for example, the character called "John" in Figure 1, would be on the play area having a picture of him sleeping, the coughing character would not infect him, because the first event in Figure 1 would not match.

The situation shown in the precondition of an event strip can be thought of as an example of a situation that

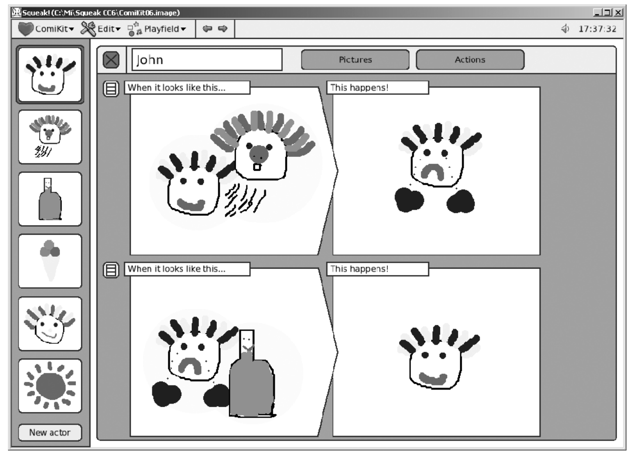


Figure 1. Programming with event strips. This example shows two events for a character called "John". In the first event strip John becomes infected and gets sick. The precondition panel shows John touching another character, who has got a cough. In the action panel, the picture of John changes to show him sad and crying (the shapes below John are pools of tears). In the second event strip, he gets cured by taking medicine. The precondition panel shows the sick John touching a bottle of medicine, and the action panel changes his picture to happy.

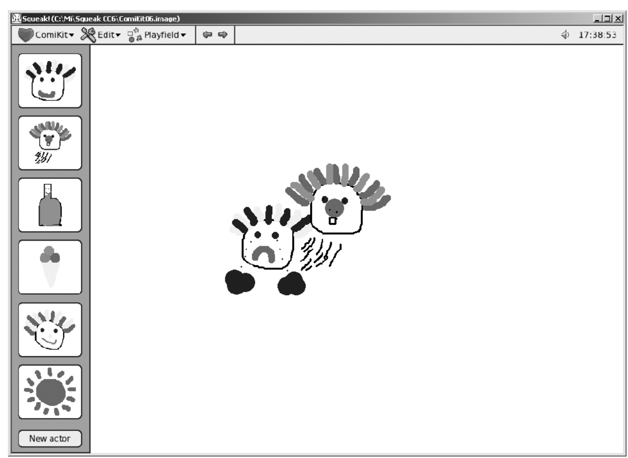


Figure 2. The play area. The player drags characters from the gallery at the left and drops them on the play area. The player can move the characters on the play area with the mouse, causing events to trigger as the characters touch each other. Here is a situation where the player has moved the coughing character with the mouse, to make it touch John. This caused the first event in Figure 1 to trigger, which changed John's picture from happy to sick.

triggers the event. On the play area, the characters need not touch each other in exactly the same way as in the precondition, for the event to trigger. Touching in any direction (left, right, above, or below), will match the precondition.

The object model that is used is similar to classes in object-oriented programming. In an event there is always a main object (similar to the notion of "self" in object-oriented programming languages such as Smalltalk). The main object is the object for which an event is defined (events belong to objects like methods belong to classes in object-oriented programming).

Dragging a character from the gallery to the play area creates an instance of that object type. Several copies (instances) of a character can be created this way (new instances can also be created in events). If there would be many copies of the character called John on the play area in Figure 2, they could all be infected by touching them with the coughing character.

The preconditions currently implemented are picture matching, touching another object, keyboard pressed, time interval, and random time interval. The actions implemented are changing the picture of an object, moving an object, delete an object, and create a new object instance. Many additional preconditions and actions could be imagined.

Several field studies of previous prototypes of the ComiKit system were made in a school together with children in grade four and five [6], [7]. The children could learn the fundamentals of comic strip programming to create programs like interactive picture-stories in one to two hours. However, basic parts of the programming model required explanation to be understood in the intended way. The main finding was that the children's intuitive interpretation of comic strip programs was that of a linear action sequence. The children had to learn programming constructs such as conditionals and concurrency to be able to successfully create programs, notions that contradicted the familiar view of comics as sequential stories.

The prototypes studied used standard rectangular panels, and the arrow-shaped precondition panel used in the current system is an attempt to make the conditional panel "look special". How this shape is understood by users has not yet been studied, however. Another problem was to put events on the "right" character. For some events it is important that the main object is properly chosen, and the children were not always aware of this.

In the prototypes used, each event could have only one action panel (this is also the case in the current system). This restriction was made primarily to simplify the implementation. Interestingly, this forced the children to break up linear action sequences into multiple events, which resulted in that the interactive stories they created could be played in a much more open-ended style than what could be expected at first [6].

3. Comic book signs and visual programming

Several comic book signs can be applied to visual programming, but there are also many symbolic signs that are needed for programming that are missing in comics.

3.1. Applying comic signs to programming

The following is a discussion of signs in comics that can be applied directly to visual event-based programs.

Characters. The appearance of a character in the program directly maps to the appearance in the runtime representation.

Panels. The panels in a comic divide time into smaller units. In a program, panels can be used to represent action sequences, and by introducing conditional panels, events with preconditions can be expressed. Note that a panel in a comic is not just a frozen moment of time. Unlike a photograph, a panel can show a situation that is extended in time. Speaking in a voice balloon, for example, is not an instantaneous action. Objects in a program could also perform multiple activities within the timespan of a panel.

Meetings. Characters in a panel are commonly shown meeting, simply by positioning them close to each other. This can be used in a program to show a collides (touches) condition in a visually direct way.

Motion markers. Comic books feature signs like speed lines and ghost images that can be used to show the motion of objects in a visual program. An example of a ghost shadow motion marker is given i Figure 3.

Transitions. In comics, as in film, cuts between perspectives and scenes are common. For example, the panels in an event strip could show objects at different places, as in Figures 3 and 4.

Inner panels. An inner panel is a panel inside another panel. Such panels are used to show something that happens at the same time, but at another place. These panels can be used by event strips to make location-independent references to objects (similar to global variable references). See Figure 5 for an example.

Voice balloons and text boxes. Balloons can be used to output text that is "spoken" by the characters in the program. Contextual text boxes could be used to represent symbolic programming constructs that are awkward or impossible to express pictorially, for example, textual and numerical object properties.

3.2 Signs that are missing in comics

The following are examples of programming constructs for which there are no corresponding signs in comics.

Conditionals. Strips in a comic book does not have conditional panels. Introducing a special sign, like an arrow between the first and the second panel, or an arrow-shaped precondition panel, could help to communicate the special status of the first panel.

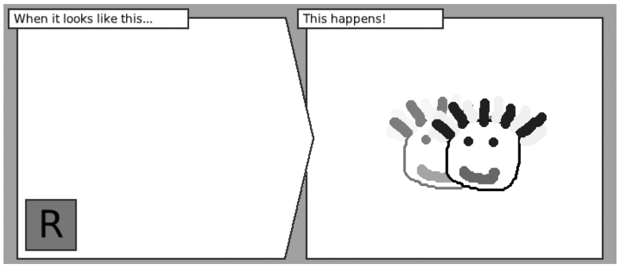


Figure 3. Motion marker. This example shows an example of a ghost image motion marker. The precondition contains a picture of a keyboard key. When pressing the R key, the character moves to the right.

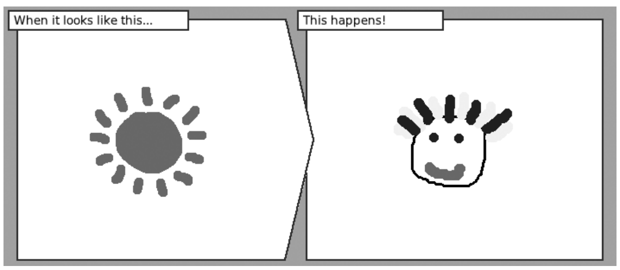


Figure 4. Panel transition. In this strip the panels show different parts of the world. When there is a sun shining in the world, the main character will become happy, regardless of its current look. This strip is an example of what is called an aspect-to-aspect transition in comics.

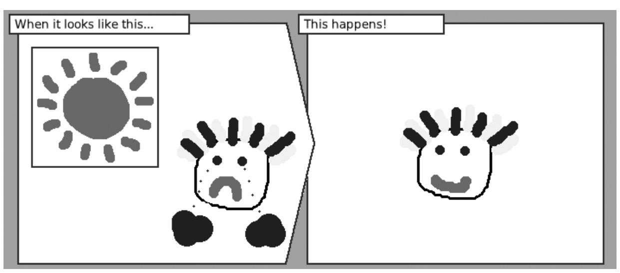


Figure 5. Inner panel. Since the Sun is inside an inner panel, it can be located anywhere for the event to trigger, but it will trigger only when the main character looks sad.

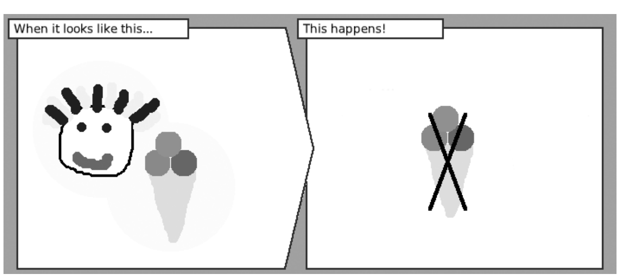


Figure 6. Delete action. In this event the main character "eats" an ice cream when touching it. The cross signifies the delete action, and is an example a contextual marker.

Non-sequential control structures and concurrency. Events in a comic strip program can be triggered in any order and execute concurrently, depending on, for instance, how the player interacts with a game. This stands in contrast to the reading order of comics, where strips represent a sequence of actions.

Deletion of objects. Comics use situations in the context of the story to show when something disappears. There does not seem to exist generic signs for deleting objects. The X-shaped cross in Figure 6 is an attempt at creating a generic sign for the delete action.

Generalisation. An example of generalisation that is handy in a precondition panel is being able to say that an object should match objects of the same kind regardless of the picture (picture generalisation), or that it should match any existing object (type generalisation). Such generalisations can reduce the number of strips needed to express an event [7]. Comics depict concrete situations and have not developed signs for these kinds of generalisations. Possible signs could be a circle with a question mark for matching any object, and an outline of the first picture of an object for matching any object picture. Note that we get picture generalisation "for free" when the main object is not present in the precondition panel (see Figures 3 and 4). However, this technique can not be used when touching other objects, because then the main object must be present in the first panel.

Variables. In a way, an object instance can be thought of as a global variable with a value range given by the pictures defined for the object type (like an enumerated data type). Objects can check the state of another object by including that object in the precondition (without touching it). However, if there are several instances of the other object, there is currently no way to identify them. Global names could be used for this, or an object could be linked to another, creating something similar to an instance variable referring to another object.

Negation. Expressing negation pictorially is another example of something that comics have no generic signs for. How would one express that an object is not touching another object, or that an object should not have a particular picture? One solution could be to negate the entire precondition panel, or to have two panels following the precondition, labelled "YES" and "NO", which would be similar to an if-else clause.

4. Beyond graphical rewrite rules

The idea of representing programs as picture sequences has also been used by systems based on graphical rewrite rules (also called before-after rules). Examples of such systems are BITPICT [8], Cartoonist [12], KidSim [13], [14], Stagecast Creator [1], and AgentSheets [11], [15], [16]. This section discusses limitations of graphical rewrite rules and how these limitations could be addressed by the visual language of comics.

4.1. Limitations of graphical rewrite rules

Graphical rewrites. Typically, graphical rewrite rules define before-after rewrites of the world state. This makes it problematic to refer to objects in a flexible way, or to objects that could be located anywhere in the world. Most systems that use graphical rewrite rules seem to have the restriction that the before part must be the same size and show the same location of the world as the after part, and except for objects created or deleted, the same objects must be present in both parts.

Iconic signs. Graphical rewrite rules consist of pictures of domain objects. This makes the representation iconic to the runtime representation, but places limits on what can be expressed. For example, in Stagecast Creator [1], symbolic program constructs are placed outside of the pictorial part of the rule.

The grid. Most graphical rewrite rule systems use a grid-based world model. There are several limitations of using a grid; objects must have fixed sizes, must be placed at discrete locations, can not overlap, and object motion is restricted and jagged. Interacting with objects that are confined to a grid in a flowing, direct manipulation style, e.g. continuously dragging an object with the mouse, is impossible. It should be noted that it is possible to use before-after rules with other models than a grid. One example is ChemTrains [9], [10], a system which uses a topological model where any topology based on containment and connected locations can be used, including a grid.

4.2. Using the visual language of comics

In the following it is discussed how the visual language of comics could be used to address some of the problems discussed above.

Panel transitions. McCloud identifies six types of panel transitions that are used in comics [3], pp. 70-72: Moment-to-moment (e.g. open eye – closed eye), action-to-action (e.g. glass being poured – someone drinking), subject-to-subject (e.g. two people talking – close up of a ringing phone), scene-to-scene (e.g. at home – at a football game), aspect-to-aspect (several aspects or moods of something, e.g. sun shining – child playing), and non-sequitur (has no obvious meaning). The transitions that are used by graphical rewrite rules are similar to moment-to-moment and action-to-action transitions where the panels show the exact same perspective. By adopting additional kinds of transitions, making before-after pictures less tightly coupled, the restrictions imposed by before-after rewrites could be relaxed. Visual programming languages could take advantage of increased expressiveness and flexibility, while preserving a direct visual mapping between the program and the runtime result. In addition to various transitions, different perspectives could be used. The precondition could for example contain a close-up of a control panel with a button in a pressed state, and the action could contain a full view of a rocket being launched.

Contextual signs. Comics mix iconic and symbolic representations, and use a wide variety of contextual signs to visualise what is going on inside the panels. Symbols, like motion markers, are used to visualise dynamic aspects within a static picture. Such markers are shown in the context of the object being modified by the sign, which creates a direct mapping between the symbolic sign and the iconic object. The tight integration of signs brings the picture to life, transforming the static image to a dynamic one in the mind of the reader. Marker signs also extend the time span of a panel, making it possible to visualise several actions within a panel.

Visual programming languages could use contextual signs to represent symbolic attributes of objects in a visually direct way. Textual and symbolic signs, like numbers and mathematical operators, could be integrated into the context of the graphical objects in a program. The way comics integrate text and pictures creates a quality of directness; the representation becomes a whole which guides the reader by providing mappings within the representation itself.

AgentSheets is a system that takes advantage of mixing textual and iconic signs [15]. However, several programming constructs in AgentSheets are text-only and rules tend to look more like illustrated texts than like pictures with text [15], [16]. The similarity between the program and the runtime result is not as strong as it potentially could be by adopting comic book techniques.

Gridlessness. Objects that live in a gridless world can be interacted with in a much more flowing way than what is possible when restricted to locations in a grid. To make direct manipulation “feel right”, the motion must be continuous. In a gridless model, objects can be any size, they can move in any direction at any speed, and they can overlap each other. Objects in a gridless model can refer to each other independently of their relative location in a grid. Inner panels are an example of how a visual programming language can use signs from comics to refer to objects in a location-independent way. Furthermore, a continuous coordinate system makes it possible to represent Logo-style turtle geometry in a visually direct way with motion markers [7].

It should be emphasised that using a grid also has several advantages, like reduced brittleness, relational transparency, clarity of spatial relationships, implicit communication between adjacent grid-cells, and regularity [11]. A gridless system could use an editing grid as a way to get some of these advantages. It should also be possible to extend a grid-based model to become more flexible, e.g. by allowing overlapping objects. A technical advantage with using a grid-based model is execution speed, in particular when checking if an object is close to objects in adjacent grid cells.

5. Conclusion

The comic strip program examples given in this paper are intended to demonstrate that the visual language of comics has the potential to represent event-based visual programs in an expressive and flexible way that directly maps to the runtime representation.

Comics have signs for expressing action sequences in a visually direct way, but lack signs for expressing conditionals, concurrency, generalisation, and other programming constructs. New signs have to found or invented to express programming concepts in a clear way, and people who want to create comic strip programs need learn these signs to become successful programmers.

An important contribution of comics is the use of contextual symbolic signs. Such signs usually have no meaning in isolation, they are an effect of what happens to the objects in a panel. Speed lines and ghost images, for example, are used to signify that something is moving. In our imagination, a character with speed lines "looks like" it is moving, even though in the medium nothing moves at all. Another example is onomatopoetic symbols, signs that imitate sound, for example the letters **Zzzzzz**, which are used to signify that someone is sleeping. In the imagination of the reader, the sign becomes the sound; seeing the sign is almost like hearing the sound. There are many other signs that once known can be said to "look similar" to what they signify, even though they are symbolic signs. Within the sign system of comics, these contextual signs take on a very direct, sometimes almost iconic, quality.

Contextual signs could be used in visual programming to visualise how objects are effected by the operations in a program. Such signs have the potential to address some of the limitations of analogical and iconic representations when it comes to expressing symbolic operations. Signs invented by comic book artists tend to become learned and accepted by the readers as they are introduced. McCloud writes: "Within a given culture these symbols will quickly spread until everybody knows them at a glance." [3], p. 131. If the comic book style of programming would catch on, a similar evolution might take place for visual programming.

In the beginning of film art, movies were like filmed theatre. A static camera recorded actors performing on a stage. Since then, film has evolved into a genre with a rich visual language. The medium of comics has evolved in parallel with film, and comic books have transferred and transformed the language of film to fit the printed medium. Visual programming with iconic representations of domain objects also has the potential to evolve, from the static camera used by graphical rewrite rules to something that we have not yet seen.

References

[1] D.C. Smith, A. Cypher, L. Tesler, "Novice Programming Comes of Age", *CACM*, vol. 43, no. 3, pp. 75-81, Mar. 2000.

[2] W. Eisner, *Comics & Sequential Art*, Florida: Poorhouse Press, 1985.

[3] S. McCloud, *Understanding Comics*, New York: HarperCollins Publishers, 1993.

[4] M. Kindborg., A. Kollerbaur, "Graphical Tools for Description of Dynamic Models", in Proc. *INTERACT'87*, Stuttgart, Germany. 1987.

[5] M. Kindborg, "How Children Understand Concurrent Comics: Experiences from LOFI and HIFI Prototypes", in Proc. *HCC'01*. Stresa, Italy, 2001.

[6] M. Kindborg, "Comics, Programming, Children, and Narratives", in Proc. *Interaction Design and Children*, Eindhoven, The Netherlands, Aug. 28-29, 2002.

[7] M. Kindborg, "Concurrent Comics - Programming of social agents by children". Ph.D. Dissertation, Linköping University, 2003.

[8] G.W. Furnas, "New Graphical Reasoning Models for Understanding Graphical Interfaces", in Proc. *CHI'91*, New Orleans, 1991.

[9] B. Bell, "ChemTrains: A Rule-Based Visual Language for Building Graphical Simulations", Technical Report CU-CS-592-92, Department of Computer Science, University of Colorado at Boulder, 1992.

[10] B. Bell, Bringham, C. Lewis, "ChemTrains: A Language for Creating Behaving Pictures", in Proc. *IEEE Workshop on Visual Languages*, Bergen, Norway. 1993.

[11] A. Repenning, W Citrin, "Agentsheets: Applying Grid-Based Spatial Reasoning to Human-Computer Interaction", in Proc. VL'93: *IEEE Workshop on Visual Languages*, Bergen, Norway, 1993.

[12] R. Hübscher, "Composing Complex Behaviour from Simple Visual Descriptions", in Proc. *IEEE Symposium on Visual Languages*, Boulder, CO, 1996.

[13] C. Rader, C. Brand, C. Lewis, "Degrees of Comprehension: Children's Understanding of a Visual Programming Environment", in Proc. *CHI'97*, Atlanta, USA, 1997.

[14] D.C. Smith, A. Cypher, J. Sphorer, "KidSim: Programming Agents without a Programming Language", in *Software Agents*, J. M. Bradshaw, Ed. AAAI Press / The MIT Press, 1997, pp. 165-190.

[15] C Rader, G. Cherry, C. Brand, A. Repenning., C. Lewis, "Principles to Scaffold Mixed Textual and Iconic End-User Programming Languages" in Proc. *IEEE Symposium of Visual Languages*, Nova Scotia, Canada, 1998.

[16] A. Repenning, A. Ioannidou, "Agent-Based End-User Development", *CACM*, vol. 47, no. 9, pp. 43-46, Sep. 2004.

[17] T.A. Sebeok, *Signs: An Introduction to Semiotics*, Toronto: University of Toronto Press, 1994.

A visual tool for multidimensional data analysis

Paolo Buono and Maria F. Costabile and Emanuele Covino and Giovanni Pani
Dipartimento di Informatica
Università degli Studi di Bari
Bari, Italy
Email: {buono, costabile, covino, pani}@di.uniba.it

Abstract

Data analysis is a crucial activity in several domains. Data analysts need tools that allow multidimensional analysis in order to select relevant information from huge quantities of data. Visualizations may help decision makers to get the information they need. In this paper we present a visual tool that supports the analysis of multidimensional data. We also provide the formal specification of its visual interface that may help designers in the implementation and may permit to easily integrate new features in the tool.

1 Introduction

Visual representations of data have the capability of shifting the load of interpreting data from the user's cognitive system to the perceptual system. In order to be understood by users, the information needs to be visualized in an information space. This visualization can either be carried out by the users in their own mind, in which case it is essentially a users' conceptualization of that information, or it could be accomplished by the system by generating a proper visualization on the display screen. The latter is defined as *information visualization*, i.e. "a process of transforming information into a visual form enabling the user to observe information" [3]. Proper visualizations allow users to easily extract patterns, trends, gaps among data.

Today the research is focused on interaction techniques that, combined with information visualization techniques, permit to reach the goal of information visualization in a more effective way. By allowing dynamic user control of the visual information through direct manipulation principles, it is possible to traverse large information spaces and facilitate comprehension with reduced anxiety. In a few tenths of a second, humans can recognize features in megapixel displays, identify patterns and exceptions, recall related images. Using proximity, color, size, animation, and user-controlled selections users are enabled to explore large information spaces rapidly and with fun.

In this paper we present DaeQP, a visual tool that supports the analysis of multi dimensional data. It exploits a visual strategy that provide data overviews in order to allow users to quickly build their mental model of the data they are interacting with and to use appropriate filters in order to analyze a reasonable quantity of relevant data.

The name comes from two reasons: 1) this tool is part of a more general framework for data analysis called Data Analysis Engine (DAE), that has been developed in the context of two projects sponsored by the European Community [2]; 2) it gets inspiration from the concept of Query Previews (QP) proposed by Plaisant et al. [5].

Query Previews allow users to rapidly gain an understanding of the content and scope of a digital data collection. The approach involves the presentation of overviews and previews of abstracted metadata enabling users to rapidly and dynamically avoid undesired data. This overcomes the inconveniences of the traditional approach to querying by a form fill-in interface, which generates user frustration when the query returns either zero hits or a very large number of hits. It is difficult to estimate how much data is available along a certain dimension and how to increase or reduce the size of the resulting set.

Compared with the prototype described in [5], DaeQP presents some novel features that make the data visualization more manageable and more general purpose, as we will show later in this paper. Moreover, DaeQP is able to get input data from any single database table, even in csv format or XML. As a further novel contribution, the visual interface is formally specified by precisely describing the semantics of the interface widgets and their functional behavior with a functional-algebraic approach. This provides a good support for designers in the implementation phase and permits to easily integrate new features in the tool.

The paper is organized as follows. Section 2 presents the DaeQP visual interface with reference to its application in the trade fair domain. Section 3 describes the formal specification of DaeQP. Conclusions are given in Section 4.

2 DaeQP

DaeQP is a tool that implements a visualization technique inspired to Query Previews [5]. Its main goal is to provide users with rapid multidimensional overviews and to allow them to perform appropriate data analysis by direct manipulation of the elements shown on the screen.

Figure 1. Different attributes along with the user can perform the analysis

The overview allow users to grasp the content of the whole dataset and their distribution across different attributes. By interacting with the overview, the user may request a query preview that provides preliminary information about data of interest, making visible problems and gaps in the data that are difficult to detect with traditional form fill-in interfaces.

Users may filter out uninteresting items and quickly focus on those of interest. Once a manageable number of items are obtained, it will be easy to browse the details about groups or individual items.

2.1 The trade fair domain

In order to understand how DaeQP works, we describe it referring to its application in the trade fair domain. DaeQP is part of a modular system supporting trade fair organization, that we have developed as part of the European funded project FairsNet (IST-2001-34290).

DaeQP is one of the visualization tools available in the FairsNet system, that have been designed and developed to present in appropriate ways the information extracted from the system repository, thus allowing the main actors of the trade fair domain to better exploit the extracted knowledge for improving their business activities. The objective of FairsNet system is to offer on-line innovative services to support business processes of trade fairs. In FairsNet, visualization techniques are used to allow the users to grasp the knowledge stored in the database, and improve human-computer interaction. FairsNet primarily addresses three types of users: fair organizers, exhibitors, and professional visitors, i.e., people who visit the fair for business reasons rather than for fun. The Data Analysis Engine functionalities aim at providing a valuable help to these users in the different phases of the decision making processes they may undergo to improve their own business.

2.2 Interacting with DaeQP

Let us consider the following scenario: an organizer of an Italian trade fair on agriculture wants to retrieve some exhibitors of the last edition of the trade fair whose data are stored in the fair database. The organizer wants to find a segment of companies, having some characteristics, to which he will send customized advertising together with the invitation to attend next event. The objective of the organizer is to increase the trade fair income by selling more services to such companies or providing services of better quality. Performing an accurate segmentation has two advantages: 1) the organizer sends material only to a reduced number of companies, thus saving costs; 2) only companies that should be really interested receive that specific material, avoiding 'junk mails'.

The organizer needs to retrieve information about Exhibitors from data stored in the fair database. In order to generate an overview in which data are visualized along some major attributes, DaeQP shows the window in Figure 1 asking the user to select one main attribute (from top pane) and some other attributes of interest (from bottom pane). The reason for selecting the main attribute is because DaeQP is domain independent and it may visualize uninteresting information if not set properly.

The attributes appearing in Figure 1 are those of the data set of interest, in our example, the exhibitors of last trade fair. Since the organizer is interested in finding exhibitors with some characteristics, he/she chooses Exhibitors as main attribute, that represents the name of the companies, and selects four other attributes as analysis dimensions to help him selecting the companies of interest. Since he is interested in addressing companies of a certain type, with a small number of employees, located in a certain geographic area and operating in some markets, he chooses the attributes Personnel, Company Type, Geographic Area, and Current Market.

The possibility of selecting several attributes is another novel feature of DaeQP with respect to the original Query Preview proposal in [5], where the number of displayed attributes was fixed to three.

The user presses the button *Next* and the interface shows the distribution of the Exhibitors along the selected attributes (see Figure 2). What appears on the screen is the overview of all the exhibitors of last year trade fair. The long bar at the bottom indicates that they are 157. The user

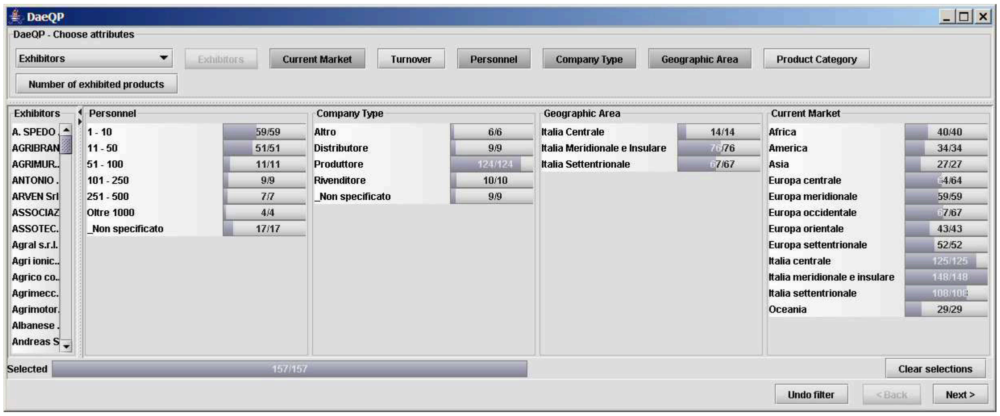


Figure 2. Overview of data along Personnel, Company type, Current Market, and Geographic Area attributes

immediately gets a lot of information. For example, only four exhibitors have more than 1000 employees, while most exhibitors have less than 50 employees and for exactly 124 the company type is Producer ('Produttore' in Figure 2).

Each bar has two numbers, the right number indicates the total number of Exhibitors having that attribute value, the left number indicates how many are selected. Therefore, in the overview in which the whole data set is shown, the two numbers are always equal. When the user makes a selection, the left number will update to indicate how many items with that attribute value are in the selected data set.

The Query Previews technique used in [5] does not provide the number on the right, but we found that it gives an important feed-back to the user. Another difference is that in DaeQP the bars have all the same length that is normalized to the maximum number of items in the whole data set, i.e. the total number of Exhibitors in our example. When each value is selected, the colored portion of the bars in the other dimensions modify to reflect the number of data that satisfy the constraints; specifically, the colored portion of the bar visually indicates the number of items with that attribute value in the selected dataset.

Starting from this overview the user may easily select companies with specific attribute values. For example, if he is interested in companies that are producers, have a few employees and operate in Northern Italy, he clicks on either the corresponding value or the bar of each attribute. The windows in Figure 3 shows the resulting data set after the user has clicked on the value "1-10" of Personnel, on the value "Produttore" of Company Type and on the value "Italia Settentrionale" (the Italian of Northern Italy) of Current Markets.

As indicated by the bar at the bottom of the Figure 3, the selected data set includes 28 exhibitors. The names are easily visible in the side bar at the left of the window, since the names of the exhibitors who are not in the data set are grayed-out.

Similarly, all values in the showed attributes but the selected ones are grayed-out to clearly indicate the characteristics of the selected exhibitors. A click on the Next button will show the list of the exhibitors in this data set. The organizer may use the set of these 28 exhibitors to send them a specific advertisement of for other purposes. If he is not happy with this selection, he can easily choose the values of the attributes or go back to the overview in Figure 2 and make different selections. The user at any moment of the interaction session may decide to display a further attribute; the attribute and its values are shown by clicking on the name of the attribute on top of the window in Figure 3 (or Figure 2). Similarly, to remove a displayed attribute, the user has to click on the attribute button.

The behavior of the DaeQP interface may be specified by a formal description that is helpful for the user interface designer because it provides both a precise and concise description of the user interface and facilitates a coherent extension of the current system with new functionalities. In the following section a formal description of the DaeQP interface in terms of functions and their semantics is given.

3 A formal specification of the user interface

Diagrammatic notations are often used to describe the interface design; examples are: state transition diagrams, Petri nets, traditional flow diagrams. Other approaches use textual notation, grammars, event algebras [4]. In this section we propose a new formalism in order to describe the structure and the behavior of the DaeQP visual interface. Our formalism is based on a typed functional approach, in

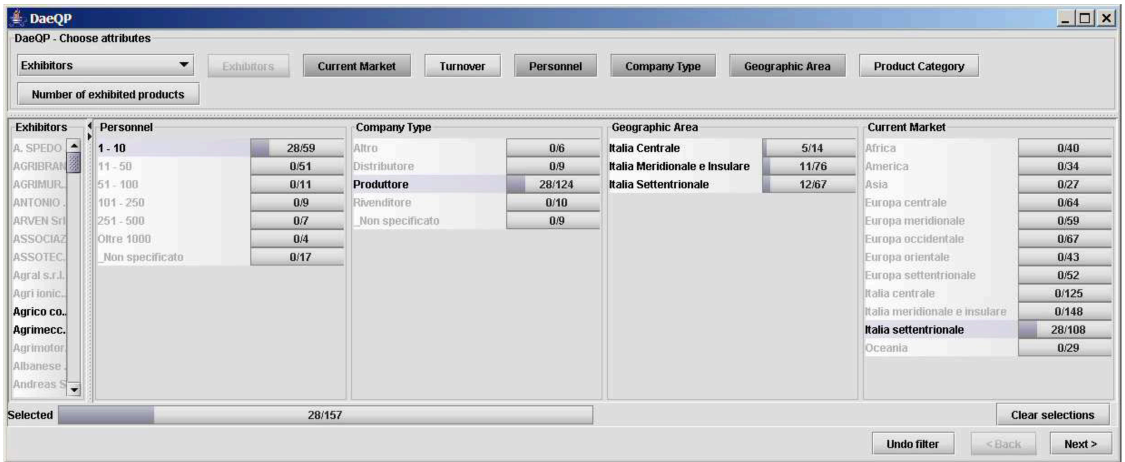


Figure 3. Query preview after a selection on values Personnel, Company Type and Current Market. The selected exhibitors are displayed in the sidebar (under the header Exhibitors)

which each variable is defined in the domain of the interface basic widgets, i.e. buttons, and each typed function represents the transition from a given state of the interface to the following one. The operational semantics of the interface is represented by the programs that manage user interaction; the denotational semantics, that is the values of the programs, represents what appears on the screen.

We have chosen a typed functional approach with the aim to provide a simpler understanding of the dynamics of the interface, which is steered by the actions of the user and by the values in the database. This approach may provide the developer with an insight of the behavior of the tool. We assume that the developer understands the formal specification.

In DaeQP the number of widgets in the screen is not predictable before the data set is loaded, and it depends also on the cardinality of the data along the dimensions taken into account. Using the more usual formalisms, such as automata, grammars, rewriting rules, etc. [1,4], the number of rules or states could be hard to manage.

In this paper a subset of our notation that works for DaeQP interface is presented; in a future work the whole formalism will be provided. In the rest of this section, we denote with:

- DB a *database*, that is a matrix of $k \times n$ elements, each of which is a string of characters denoted with $DB[i, j]$;
- Π_j the ordered sets $\bigcup_{i=1...k} DB[i, j]$, for each $j = 1 \ldots n$;
- $Button$ the set of buttons $\{A_1, \ldots, A_n\}$;
- $S := \{s, d, *s, *d\}$ the set of the button states;
- f_i a *flag*, that is an element of S;
- B the set of all couples ($button$, $flag$).

Definition 1 *A sequence of attributes is an n+1-ple* $p := (A_a, (A_1, f_1), \ldots, (A_n, f_n))$*, where*

1. A_a *is an element of* $Button$*, called combobox; all the* $A_i, i = 1, \ldots, n$ *are different elements of* $Button$*; and there is at most an* A_j *such that* $A_a = A_j$*;*
2. $f_i \in \{s, d\}$*, for all* $A_i \neq A_a$*,* $i = 1, ..., n$*;* $f_i \in \{*s, *d\}$ *otherwise.*

As an example, at the top of Figure 2 a sequence of attributes is shown: the top left button Exhibitors is the combobox, all the others are couples $(Button, flag)$ some of them have the flag switched to s (selected), namely Current Market, Personnel, Company Type, and Geographic Area.

The button states should be understood as follows: if the flag associated with a button A is equal to s (resp., d) then A is a selected (resp., deselected) button; if the flag associated with a button A is equal to $*s$ (resp., $*d$) then the combobox is equal to A and the button A was in the state s (resp., d) before it was chosen as a combobox. Note that when the user will select a different combobox, the flag of the second occurrence of A_a must be changed to s (resp., d).

At the beginning of the computation the user selects a subset of the buttons and a combobox, setting the *initial configuration* of the sequence of attributes (see Figure 1).

Once the user has selected a sequence of attributes, what appears on the screen will be defined as the "denotational semantics" of that sequence.

Definition 2 *Given a sequence of attributes* $p = (A_a, (A_1, f_1), \ldots, (A_n, f_n))$*, the denotational semantics of* p*, denoted with* $[p]$*, is the sequence* $(\widehat{\Pi}_a, \widetilde{\Pi}_1, ..., \widetilde{\Pi}_n)$ *where*

1. $\widehat{\Pi}_a = \{(element, flag) \in \Pi_a \times \{s, d\}\}$*, and*

2. $\widetilde{\Pi}_i = \begin{cases} \{(element, flag, count) \in \Pi_i \times \{s, d\} \times N\} \\ \textit{if } f_i = s \\ \emptyset \textit{ otherwise} \end{cases}$

The elements of $\widehat{\Pi}_a$ are attributes composed by two components: the first is an element (a string) of Π_a, and the second is a flag. The elements of $\widetilde{\Pi}_i$ are attributes composed by three components. The first one is a string belonging to Π_i, the second is a flag, and the third is a natural number called $count$. The flags of these new buttons can assume only the values s or d. At this point of our description, given a sequence of attributes p, we can have different denotational semantics, that have different values of flags and counts. We see that the choice of elements, flags and counts will became unique depending on the initial configuration (the first choice of the sequence of attributes), and on the choices of combobox and attributes that the user will do during the work session.

Given a sequence of attributes p, the *view* (or interface) of p is the couple $(p, [p])$, denoted with $[[p]]$. Thus a view is given by a list of attributes and the related denotational semantics. The set of all the views is denoted with V (Figure 2 and 3 are examples of interface).

Given a couple $a_i := (A_i, d)$, we denote with a'_i the couple (A_i, s); and given a couple $a_i := (A_i, s)$, we denote with a'_i the couple (A_i, d). With α and β are denoted elements of the set $\{s, d\}$.

To describe how it is possible to transform a view into a new one, it is indispensable to define both the users' activity and the functions that have to be performed and computed to obtain the new view. In our system two different cases are possible when the user clicks an element in the sequence of attributes. The first case happens when the user clicks an attribute that is not the combobox. The second happens when the combobox is clicked and a list of buttons appears, and then the users clicks on an element of that list. The types of the functions that change the sequence of attributes, are denoted with either $A \times Button \to A$ or $A \to (Button \to A)$. In our notation the meaning of a function f with type $A \times Button \to A$ should be intended as follows: given $v \in A$ and $(A_k, \alpha) \in v$, f defines a new view $v' \in A$. From the interface point of view, this means that a menu with a sequence of attributes (a sequence of buttons) is presented to the user, and he/she can change the state of one of them, defining a new view. The meaning of a function g with type $A \to (Button \to A)$ should be intended as follows: given $v \in A$, the function g defines a function of type $(Button \to A)$, and choosing an element of $Button$ (which means that a list of buttons appears and the user clicks one of its elements) a new view $v' \in A$ is defined. In our case only the combobox has a list of elements, then it is not necessary to define a function that associates attributes to their list. Note that if we have a function $h : A \times B \to A$ such that all the elements of B are elements of A, it is not possible to define a $h' : A \to (B \to A)$ such that $h = h'$.

Given a new combobox value (a new A_a), the following function produces a new sequence of attributes (the user changes the combobox value).

Definition 3 *A change-of-combobox is a function* $f : A \to (Button \to A)$ *such that*

1. *if* $(A_j, a_1, \ldots, a_h, \ldots, a_j, \ldots, a_n) \in A$*;* $a_h = (A_k, \alpha)$*; and* $a_j = (A_j, *\beta)$*; then* $f((A_j, a_1, \ldots, a_k, \ldots, a_j, \ldots, a_n)(A_k)) = (A_k, a_1, \ldots, a'_k, \ldots, a'_j, \ldots, a_n)$*, where* $a'_h = (A_k, *\alpha)$*, and* $a'_j = (A_j, \beta)$*;*

2. *if* $(A_j, a_1, \ldots, a_j, \ldots, a_n) \in A$*;* $a_j = (A_j, *\beta)$*; and all the* a_i *are different from* (A_k, α)*; then* $f((A_j, a_1, \ldots, a_j, \ldots, a_n)(A_k)) = (A_k, a_1, \ldots, a'_j, \ldots, a_n)$ *where* $a'_j = (A_j, \beta)$*.*

Note that if an attribute with the same name of the old combobox occurs in the sequence, its flag $*\alpha$ is changed to α; and if an attribute with the name of the new combobox occurs in the sequence, its flag α is changed to $*\alpha$. The following function describes what happens when the user changes the state of an attribute (the user clicks a button).

Definition 4 *The change-of-button-state is a function* $g : A \times B \to A$ *such that* $g((A_j, a_1, ..., a_k, ..., a_n), A_k) = \begin{cases} (A_j, a_1, ..., a'_k, ..., a_n) & \textit{if } k \neq j\textit{ ;} \\ (A_j, a_1, ..., a_k, ..., a_n) & \textit{otherwise.} \end{cases}$

In DaeQP the elements of denotational semantics $[p]$, assume the same role of a generic attribute; thus, they can be selected or deselected. We describe what happens when we select a button in $[p]$.

The values of $count$ (see definition of denotational semantic) are defined by means of a set of standard statistic functions, denoted with γ_j^{DB}, which depend on the contents of the database and on the current sequence of attributes. In this paper we are not interested to define their operational semantics.

Given the set $\widehat{\Pi}_a$; a couple $(B, flag_B) \in \widehat{\Pi}_a$; and a couple $\delta := (B, \overline{flag_B})$; we denote with $\widehat{\Pi}_a + \delta$ the set obtained by substituting $(B, flag_B)$ with δ.

Given a set $\widetilde{\Pi}_i$; a triple $(B, flag_B, count) \in \widetilde{\Pi}_i$; and a triple $\delta := (B, \overline{flag_B}, count)$; we denote with $\widetilde{\Pi}_i + \delta$ the set obtained by substituting $(B, flag_B, count)$ with δ. We will use these notations when the user selects or deselects a button.

Given set $\widetilde{\Pi}_i$; with $\widetilde{\Pi}'_i$ is denoted the set $\{(element, flag)\ |(element, flag, count) \in \widetilde{\Pi}_i\}$.

The following functions define the selection/deselection operation performed on an element of $\widehat{\Pi}_a$ or of $\widetilde{\Pi}_i$, with $i = 1, ..., n$; and the change of values of all the $count$'s (by means of the functions γ_i^{DB}) and of the flags (by means of the functions $ch - flag$) .

As we have written before, we don't define the operational semantics (the programs) of functions $ch - flag$ and γ_i^{DB}, but only where their results are visualized on the interface. The γ_i^{DB} and the $ch - flag$ functions have as input a view and an attribute and as output a new count and new flag, respectively.

Definition 5 *The combobox filter-function* $\Phi_a : V \times \widehat{\Pi}_a \to V$ *is such that, given* $v = (A_a, A_1, \ldots, A_n) \in V$ *and* $\delta = (b, \overline{flag_b}) \in \widehat{\Pi}_a$*, we have:* $\Phi_a((A_a, A_1, \ldots, A_n), \delta) = (B, B_1, \ldots, B_n)$*, where* $B = \{(element, ch - flag(v, \delta))|(element, flag) \in A_a\}$ *, and* $B_j = \{(element, ch - flag(v, \delta), \gamma_i^{DB}(v, \delta))\ |(element, flag) \in \overline{A_j}\}$*, with* $j = 1, \ldots, n$*.*

Then, given a view and an attribute in $\widehat{\Pi}_a$, the combobox filter function changes the flags and the counts. The function will be used when the user "switches" a flag of an attribute in $\widehat{\Pi}_a$. In the same way we define filter-functions that change flags and counts when the user switches an attribute in Π_i. The filter-functions $\Phi_i : V \times \widetilde{\Pi}_i \to V$ (with $i = 1, \ldots, n$) are such that, given $v = (A_1, A_2, \ldots, A_n) \in V$ and $\delta = (b, \overline{flag_b}, count) \in \widetilde{\Pi}_i$ we have that $\Phi_i((A_a, A_1, \ldots, A_n), \delta) = [[(B, B_1, \ldots, B_n)]]$, where $B = \{(element, ch - flag(v, \delta))|(element, flag) \in A_a\}$, and $B_j = \{(element, ch - flag(v, \delta), \gamma_i^{DB}(v, \delta))\ |(element, flag) \in \overline{A_j}\}$, with $j = 1, \ldots, n$.

The *transition* from one view to another can be obtained by applying either the change-of-combobox function, or the change-of-button-state function, or by applying a filter function, or the combobox filter function. Then, given $[[p_1]] = (p_1, [p_1])$ and $[[p_2]] = (p_2, [p_2])$, $[[p_2]]$ is obtained by a transition from $[[p_1]]$ (denoted with $[[p_1]] \triangleright_j [[p_2]]$) if there exists $j \in \{1, \ldots, n\}$ such that $f(p_1, A_j) = p_2$ (change of combobox); or there exists $j \in \{1, ...n\}$ such that $g(p_1, A_j) = p_2$ (change of attributes); or there exists $\delta \in \Pi_a$, such that $[[p_2]] = \Phi_a([[p_1]], \delta)$ (select/deselect an element of Π_a); or there exists $\delta \in \Pi_i$, and $[[p_2]] = \Phi_i([[p_1]], \delta)$ (select/deselect an element of Π_i). Eventually, a *session* between the user and the interface is a sequence $[[p_1]] = (p_1, [p_1]), \ldots, [[p_m]] = (p_m, [p_m])$ such that $[[p_1]] \triangleright \ldots \triangleright [[p_m]]$ where all the flags in $[[p_1]]$ are s(elected) and the $count$'s values are computed by the functions γ_j^{DB}. At the beginning of a session, the user chooses the attributes of his/her interest, then all the elements of the view are defined with their flags s(elected) and the counts computed. The session continues selecting and deselecting elements (buttons or attributes) of the view.

4 Conclusions

Many activities in decision making processes involve the selection and the analysis of a reduced number of data of interest. Often this reduced data set is difficult to get. We have shown how this goal may be reached by DaeQP, a visual tool that supports the analysis of multidimensional data. The work on DaeQP has room for improvements. In order to allow the user to improve the filtering mechanism and to get more details it would be useful to have the possibility to manage web addresses and images. We are also working on DaeQP scalability: currently DaeQP manages a few thousands of records, but we would like to manage bigger quantity of records.

The proposed formal specification is able to describe unambiguously the behavior of DaeQP. Moreover it helps the designer to easily extend the functionalities of the user interfaces, making sure that they will not be in contrast with the previous behavior. This happens because once defined the behavior of the system (its syntax and semantics), we can introduce new widgets or features simply adding their semantics, without changing the already defined semantics.

References

[1] P. Bottoni, M. F. Costabile, and P. Mussio. Specification and dialogue control of visual interaction through visual rewriting systems. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 21(6):1077–1136, 1999.

[2] P. Buono. *A framework for Visual Data Analysis*. PhD dissertation, Università degli Studi di Bari, Dipartimento di Informatica, 2005.

[3] S. K. Card, J. D. Mackinlay, and B. Shneiderman. *Readings in Information Visualization*. Morgan Kaufmann Publisher, San Francisco, California, 1999.

[4] A. Dix, J. Finley, G. Abowd, and R. Beale. *Human-computer interaction (3rd ed.)*. Pearson/Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2003.

[5] C. Plaisant, B. Shneiderman, K. Doan, and T. Bruns. Interface and data architecture for query preview in networked information systems. *ACM Transaction in Information Systems*, 17(3):320–341, 1999.

WGBUILDER: a Visual Environment for Building Web GIS Applications

Luca Paolino, Monica Sebillo, Genny Tortora, Giuliana Vitiello
Dept. Matematica e Informatica, University of Salerno, Italy
email {lpaolino, msebillo, gvitiello, tortora}@unisa.it

Abstract

Web GIS development often forces application domain experts to learn programming languages and complex frameworks adopted, thus reducing their productivity. As a solution to this issue, we present a visual environment, named WGBuilder, which supports unskilled users in the construction of Web GIS applications. By performing a set of predefined actions, users specify the customization requirements and the resulting application can be downloaded either for local use or for use on a different web server. Thus, users are only required to have knowledge about data featuring the application under construction.

Keywords
Visual environments, geographic data, GIS applications, geographic metadata.

INTRODUCTION
The impact of the Internet on GIS (*Geographical Information Systems)* is notably affecting users' activities. Application domain experts are evolving towards a wider and demanding community, whose interests are mostly directed to the dissemination of information and to advanced processing tools. Experts require GIS functions able to conduct spatial analysis and derive new information in specific domains, without having to afford expensive investments [1, 2].

In recent years, increasing attention has been devoted to the development of components for both conducting on-line GIS analysis and disseminating derived results.

L. Paolino is with the Dipartimento di Matematica e Informatica, University of Salerno, Italy (phone: +39089963324; fax: +39089963303; e-mail: lpaolino@unisa.it).
M. Sebillo is with the Dipartimento di Matematica e Informatica at the University of Salerno, Italy (msebillo@unisa.it)
G. Tortora is with the Dipartimento di Matematica e Informatica at the University of Salerno, Italy (e-mail: tortora@unisa.it).
G. Vitiello is with the Dipartimento di Matematica e Informatica at the University of Salerno, Italy (gvitiello@unisa.it)

The aim of this research consists of allowing users to build and integrate simple customized applications in agreement with their needs, by accessing providers distributed on the web. This is achieved thanks to an advanced software framework, able to generate customized Web GIS applications, where users' data and spatial features can be handled and queried through conventional mapping tools. Such a requirement arises from the need to disseminate and share geographic information through applications whose contents may be subject to frequent updates, upon user's manipulation and gained results. For that reason, an application which only visualizes the geographic content is not sufficient. It is necessary to develop an efficient and effective tool which enables users to visualize, query, analyse and process geographic data.

The traditional approach to Web GIS application development has been to implement *ad hoc* software solutions using both programming languages and Application Programming Interface (API) that manage spatial data. Available APIs were used to build platform-independent customized applications. APIs allowed to design several kinds of applications, such as simple viewers and tools to query both spatial and alphanumeric data. Examples of this approach to GIS application development are the ORACLE Spatial API, the FME object by Safe Software and the ESRI MapObject, which allow users to develop Java and COM applications.

More recently, major GIS software companies have developed specialized software for web application development. As a matter of fact, ArcIMS by ESRI [4], MapGuide by AUTODESK [3], and WebMap by INTERGRAPH [5], can be used to build web applications which allow users to query and manipulate spatial information through browsers, such as Internet Explorer and Firefox.

However, one problem with the use of GIS technology is that, in spite of the high potentiality offered, users are often faced with the implicit complexity of data and are forced to become familiar with related concepts. The development of Web GIS applications is no exception, since final users are supposed to be either domain experts, who have usually low familiarity with GIS technology, or ordinary people interested in browsing data of interest.

The project presented in this paper represents a solution. It consists of a visual environment, named *WGBuilder* (*Web GIS Builder*), which allows unskilled users to build a self contained customized Web GIS, starting from a given geographic dataset. Performing a predefined set of actions, users can select and visualize map layers from a cartographic database, can choose attributes on which queries may be posed, and can set additional details about the graphical interface. The resulting application composed by a Java applet, HTML and Jscript codes represents a GIS component, which can be either put on the Internet in order to share and disseminate its geographic content, or integrated with other components to assemble a more complex one.
The rest of this paper is organized as follows. Section 2 describes the underlying architecture of *WGBuilder*, as well as some details about the implementation. In Section 3 an example of a Web GIS generation is depicted for expert users in the environmental domain. Final remarks conclude the paper.

THE WGBUILDER VISUAL ENVIRONMENT

In the present section we describe the architecture of *WGBuilder* and tasks required to generate a Web GIS application. *WGBuilder* is based on a client/server architecture and is made up of 3 components, handled by the *WGBuilder* developers, as shown in Figure 1. Besides the basic components, such as the application server and the web server, the server side contains also the cartographic database. Each map is organized as a shapefile (.shp) and is then associated with a .dbf file, representing the descriptive part of a geographic layer. *WGBuilder* outputs a Web GIS application in terms of a Java applet, the construction of which starts when the user connects to the environment. In particular, as a first step, user's requirements are acquired through a set of pages, based on the JSP technology.

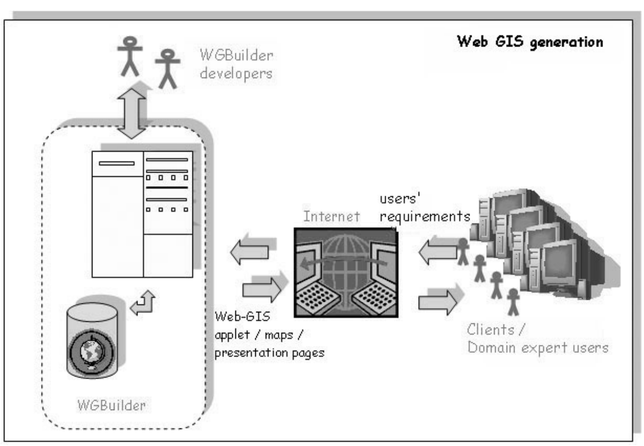


Figure 1. WGBuilder architecture

Filling some forms, the user communicates with the server and sends it the project name, as well as some interface customization details in terms of feature colours and background. The user is also required to select the attribute set on which Web GIS users may pose queries. The attribute list is visualized through a HTML and Jscript page.
Finally, based on all those parameters, the servlet builds the customized Web GIS application.
Once the application is generated, it can be either downloaded for local use, or put on a different web server to disseminate and share the geographic information it contains.

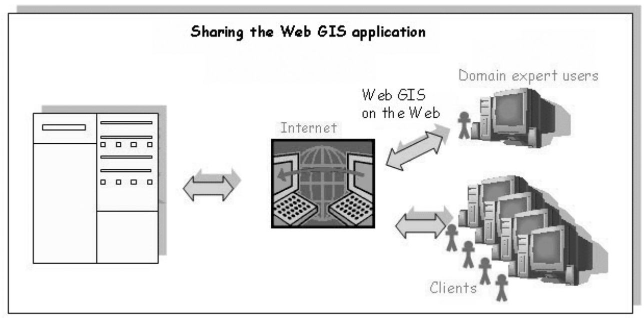


Figure 2. Sharing the Web GIS application

The user may also choose to download (for local use or for sharing) simple maps as shapefiles rather than the whole project as a Java applet together with the presentation pages as .html files. Figure 2 shows such a task.
Users may interact with the underlying database according to two approaches. Basically, they can select some of the available maps and build the Web GIS application on them. On the contrary, if the database doesn't contain the required map, users can add them to the database and then invoke the construction of the application. The provision of a map has to follow some predefined steps and the map itself has to be organized in agreement with a predefined format in terms of attributes and features. Typically, either a vector file or a raster file may be provided, together with a set of metadata which contribute to specify the quality of the map characterizing the Web GIS under construction. Table 1 reports the set of metadata with which .shp files and tiff/jpeg/bmp files are associated. They refer to the International Standard ISO/FDIS 19115 [6] and are distinguished in mandatory and optional metadata, based on their relevance when a quality value is calculated.
Finally, users who provide the cartographic database with new maps, may decide whether to physically share them with other users. If so, their files appear among the available maps and can be used in different Web GIS constructions, otherwise only a preview is available together with the information about the owner.

Attribute	Description	Domain
ID *	Cartographic identifier	Free text
Title *	Name by which the cited resource is known.	Free text
Language	Language used within the dataset	Predefined values (ISO 639-2)
Date*	Reference date for the cited resource	YYYY/MM/DD (ISO 8601)
Coverage	The extent or scope of the content of the resource	Predefined values
Keyword*	A topic of the resource content	Free text
Author*	An entity primarily responsible for making the content of the resource	Free text
Abstract	Brief narrative summary of the content of the resource	Free text
Format and version*	Description of the format of the data to be distributed	Predefined values
Resource location	Location where the resource can be looked up	Free text
Contributor	An entity responsible for making contributions to the resource content	Free text
Copyright	Information about rights held in and over the resource.	Free text
Responsable party	Identification of, and means of communication with, person(s) and organization(s) associated with the resource	Free text
Geographic location of dataset	Extended information including the bounding box, bounding polygon, vertical, and temporal extent of the dataset	Free text
Reference system*	Name of the reference system	Free text
Scale	Factor which provides a general understanding of the density of spatial data in the dataset.	Free text
On-line resource	Information about (online) sources from which the resource can be obtained	Free text
Character set	Full name of the character coding standard used for the dataset	Predefined values
Spatial representation type	Method used to spatially represent geographic information	Predefined values

Table 1. Metadata for map description – an asterisk indicates a mandatory metadata

WGBuilder and its resulting applications involve and interact with different typologies of users. As depicted in Figure 1, clients involved in the generation of Web GIS are mostly domain experts, who wish to share the information contents. They usually need to implement Web GIS applications in order to disseminate their cultural heritage through the Internet. Notwithstanding their low familiarity with computer technology, they are able to build a stable and performing application, thanks to the support of the underlying toolkit. Differently, in Figure 2 also generic clients are depicted, who interact with GIS applications put on the Web. They may be either end users interested in the information browsing or expert users who exploit shared components and data for further processing.

THE GENERATION OF THE WEB GIS PAPriCa

In the present section we show the effectiveness of the proposed visual environment in generating customized Web GIS applications starting from geographic databases. In particular, the present example shows the construction of a Web GIS containing data about reserved areas and natural parks in the Campania Italian region. The resulting application, which is named PAPriCa, (**P**rogetto **A**ree **Pr**otette **i**n **Ca**mpania – Project of Reserved Areas in Campania), allows users to acquire information about all the Campania zones, where environmental constraints have been set and the life quality is monitored.

Figure 3 shows the environment as it appears when *WGBuilder* is invoked. Users are provided with a preview of available maps, each one associated with the corresponding descriptive component. Thanks to the use of a GUI, they can intuitively manipulate both spatial and thematic components of geographic data, in a uniform way.

Figure 3. WGBuilder environment

Once the vector file containing the boundaries and the typology of reserved areas have been selected, the project name *PAPriCa* can be provided and attributes that will compose the Search frame can be selected. Also, more details about colours of layer and background can be set.

At this point, the generation of the expected Web GIS may start. Figure 4 illustrates the resulting application as it appears to the user. It consists of three frames, namely the *Search* area, the *Map viewer* and the *Details* area. In particular, the Search area provides users with the set of selected attributes on which textual queries based on conditions can be posed. Features satisfying the conditions are highlighted on the map.

The Map viewer contains a set of buttons which allow users to handle conventional mapping tools, as well as to perform spatial selections based on graphic tools. The Details area may contain links to different multimedia files associated with features selected by queries. On demand, such files can be opened to visualize videos, images and descriptions.

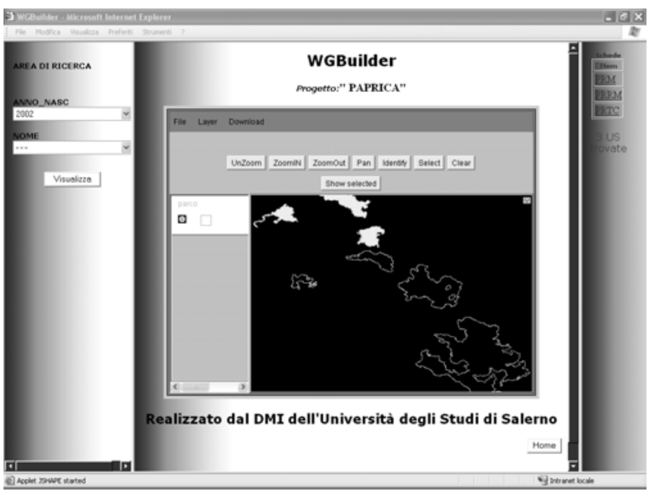


Figure 4. The resulting Web GIS

Finally, through the Download function the user can choose whether to get the whole project (applet and .html files) or to acquire only geodata as .shp files. Figure 5 shows the interface to accomplish such tasks.

FINAL REMARKS ON WGBUILDER

In the first twenty years GIS were exclusively used by programmers who mainly interacted with the system using command lines and macro languages. Recently, high potentialities of such systems were discovered by new users working in any life sectors. Consequently, new interaction necessities arose and dealing with usability problems is today considered a crucial activity in the development of GIS software, with notable impact on the overall costs.

WGBuilder has been conceived with the aim of reducing usability design costs. A user who wishes to build his/her own Web GIS, learns quickly the steps he/she is supposed to perform in order to accomplish the generation task. He/she navigates through a sequence of intuitive graphical interfaces where he/she may retrieve maps he/she is interested in, and then specify parameters needed to customize the Web GIS application.

Figure 5. The download functionality

Other features contribute to make *WGBuilder* an effectively usable toolkit. It is distributed, and searchable, that is to say, it is available on the network and can be invoked on demand. Due to the interoperability of Java, *WGBuilder* is also open and can be integrated with other components. Finally, a Web GIS built through *WGBuilder* represents a self-managing component. Users should have knowledge only about data contents and possibly about the modality to put an application on a web server.

REFERENCES

[1] Longley PA, Goodchild MF, Maguire DJ, Rhind DW. *Geographical Information System Principles* - Second Edition 2001. Wiley.

[2] Longley PA, Goodchild MF, Maguire DJ, Rhind DW. *Geographical Information Systems and Science* 2001. Wiley.

[3] AUTODESK. MapGuide, usa.autodesk.com/adsk/servlet/index

[4] ESRI. ArcIms - www.esri.com/software/arcgis/arcims/index.html

[5] Intergraph. WebMap, imgs.intergraph.com/gmwm/

[6] *ISO/FDIS 19115*, http://www.ncits.org/ref-docs/FDIS_19115.pdf

Proceedings

The 2005 International Workshop on Mobile Systems, E-commerce and Agent Technology (MSEAT 2005)

Co-Editors

Jason C. Hung, Northern Taiwan Institute of Science and Technology, Taiwan
Gwo-Jong Yu, Aletheia University, Taiwan

Computers & Security

Secure business application logic for e-commerce systems

Faisal Nabi

Internet Security Group,Department of Computer Science ,Colorado Sate University
Fort Colins,Co 80523-1873,USA

KEYWORDS
E-commerce;
Security;
Privacy;
Client trust;
Business application logic;
SSL;
CGI scripts

Abstract The major reason why most people are still sceptical about e-commerce is the perceived security and privacy risks associated with e-transactions, e.g., data, smart cards, credit cards and exchange of business information by means of online transactions. Today, vendors of e-commerce systems have relied solely on secure transaction protocols such as SSL, while ignoring the security of server and client software. This article, *Secure Business Application Logic for e-commerce Systems*, discusses a key weak link in e-commerce systems: the business application logic. Although the security issues of the front-end and back-end software systems in e-commerce application warrant equal attention, but this research focuses on the Security of *Middle Tier* of e-commerce server that implements the business application logic and traditionally, e-commerce sites implemented the middle tier of software on the web server using CGI. We also present strategies for secure business application logic: good design and engineering, secure configuration, defensive programming and secure wrappers for server-side software.

Introduction:

A risk constitutes something negative or bad. Business people, however, need to think of risks as mere opportunities, the reason being that, in most business environment, the number or size of the risks taken usually is equal to the number or size of the advantages to be gained. The reverse is also true. Namely, the number or size of the risks taken usually also equals the losses potentially to be sustained. It is, therefore, important always to take calculated risks and to be aware of all possible consequences (Labuschagne and Eloff, 2000).The advent of the electronic commerce ushered in a new period pervaded by a sense of boundless excitement and opportunities. Although it took some time, but now most organisations have already involved conducting electronic commerce as that world wide e-Commerce Market has exceeded $46 billion in consumer transactions by the year 2001 which became main focus of organization involving into e-commerce (Ernst and Young,1998).''Electronic Commerce'', also referred to as ''e-commerce''has revolutionized the modern-day business world. Thanks to its concomitant technologies, new business opportunities have been created. This results in the survival or downfall of many organizations on the global economic playing field, depending on whether they choose to seize or fail to avail themselves of these opportunities. The new opportunities, however, come with their own set of problems. The major concern cited by most decision-makers when it comes to e-commerce,is''Security, Privacy and Client Trust''. For this reason, subscribers still feel uncomfortable about the idea of trading on the Internet (Foresight, 1998).To them, the possible risks to be incurred justify the potential rewards. Unfortunately, their fears are not unfounded because of trust, privacy and security threats to all e-commerce transactions. Basically, e-commerce is concerned with doing business using electronic technologies. It can involve the transaction of data, transaction of payments or marketing information, and value addition to existing products or databases. E-commerce can involve the exchange of credit card numbers that represents purchases by a consumer from a retailer (Lawrence and Corbett, 2000).Credit card is one of the primary means of electronic payment on the Internet. Although a large number of users reported that they had their credit card stolen, there is still a lot of consumer confidence in the use of credit card. Again, this trust should not be betrayed and arrangements should be made to assure that consumer confidence remains strong in credit card specially for those who are reluctant (Kalakos and Whinston, 1996).According to the survey report by Bank of England on 15-10-2002, this year the credit card theft worth was four hundred and eleven million pounds (fraud of credit cards), which was a 50% increase compared to the previous year in the UK (Survey Report by Bank of England, 2002).Another case reported in 2003 where the number of user of Barclay Bank suffered while conducting online credit card transaction this time was the case of during the transaction client to server side, actually it was the case of ''Web Copy Cat Trick''. This technique is the cause of misconfiguration server side components that provided path way to intruder to play a technique which is known as ''Web Copy Cat Trick'' and organisation (Barclay Bank) suffered heavy loses due to that, they had to advise their customers not to use online services until problem is resolved. Another cyber crime case reported by MSNBC news report: the issue of international credit card thievery and fraud burst into the public consciousness in January 2004 with news of a heist of thousands of credit cards from the CD Universe Web site, allegedly by a teenage hacker from Russia. It continued to make headlines with news that Amazon.com had uncovered a Russia-based plot to defraud it and other e-merchants out of more than $70,000 worth of merchandise using stolen credit-card information. Stephen Orfei, vice president for electronic commerce and emerging technology at Master-Card, said that the Internet accounts for between 2 and 2.5 percent of total credit card transactions. Using the lower figure and applying it to the bank association's total fraud losses of $526 million in 1998 yields a loss to online fraud of slightly more than $10.5 million. Visa USA reported $487 million in fraudulent charges last year, which assuming a 2 percent rate would be approximately $9.75 million. American Express and Discover Financial Services do not publicly report fraud rates and declined to comment on the subject of Internet credit card fraud except to say that they aggressively attempt to comb. Since e-commerce fraud rates are one-third higher than overall fraud rates, according to Visa's figures for 1999 (9 cents per $100 on the Net vs. 6 cents per $100 overall), that $20 million figure for Visa and MasterCard alone is most likely on the low side. The federal government's main mechanism for tracking economic crimes is the Treasury Department's Financial Crimes Enforcement Network,known as FinCen, which collects reports on 17 types of ''suspicious activities'' from U.S. banks and other institutions, the agency received about 120,000 reports, of which 4900 e or about 4 percent e reflected possible credit card fraud. Increasing ratio of e-commerce fraud rates is real concern.

Concerns of e-commerce

Any business that is still in business, conducts ''commerce''. Commerce is the exchange of money for goods or services between companies (B2B) or end consumers (B2C). ''E-commerce''is doing commerce using electronic technology such as intranets, extranets and the Internet, which, provides with a new means of obtaining useful information and purchasing products and services between companies and end consumers. Although this form of e-commerce has undergone rapid growth, particularly through the use of the Internet, business and consumer fears and concerns about the risks, both have inhibited its growth real and perceived, of doing business electronically (Kamthan, 1999).From the time a business installs a web server or hires space on a commercial web server from an ISP, there is a potential for business systems in the organisation to be exposed to breaches of security and confidentiality across the entire Internet (Lawrence and Corbett, 2000).Any link to the Internet exposes the business to tampering, or Internet graffiti, where data can be exposed with meaningless scribble, pictures or electronic junk in the same way that graffiti artist scrawls on walls. Link to the Internet also exposes the business to the theft of data. Database can be very easily captured wholly and transferred for other uses such as industrial espionage (Sielgel,1996).The TCP/IP protocol developed to run the Internet was not designed with security in mind. This protocol, the basic system running Internet communications, is vulnerable to interceptions (Hunt, 1998).Any movement of data from a browser to a server or back is vulnerable to eavesdropping (Chaffy, 2000).Website security is about keeping strangers out, but at the same time allowing controlled access to the network (Lawrence and Corbett, 2000).Therefore, due to all these reasons, newer schemes for security, such as Secure Socket Layer (SSL), have been introduced. This is a low-level encryption scheme used to encrypt e-commerce communications and for consumers to use their credit card numbers in transactions on a secure WWW form and transmit the form over the Internet without the risk of a cracker obtaining the credit-card information in higher-level protocols such as SHTTP, NNTP, and FTP.The SSL protocol can authenticate server (i.e., verify server's identity) encrypting data in transit between the server and client accessing the server (Ahuja, 1997).Basically, cryptography is based on encryption, which use the authentication and digital signature techniques based on Cryptographic Methods. Encryption is the transformation of data into a form unreadable by anyone without a secret key. Its purpose is to ensure privacy by keeping the information hidden from anyone for whom it is not intended, even those who can see the encryption data. There are two types of cryptographic systems:(1) symmetric system, where a single key serves both as the encryption and decryption; (2)asymmetric system, where each person gets a pair of keys called the public and private key. Each person's public key is published, while the private key is kept secret. The need for the sender and receiver to share secret information is eliminated.All communications involve only public keys, & no private key is ever transmitted or shared.Furthermore, public and private key cryptography can be used for authentication using Cryptographic Envelop Technology (which is developed in Java Archive) technique (Lawrence and Corbett, 2000).This session key is used to encrypt the HTTPtransaction (Furche and Wrightson, 1996).Nevertheless, the advancement of the security field has proved that vendors of e-commerce systems cannot solely rely on secure transaction protocols, such as SSL e an encryption protocol promoted as proof of 100% security by e-commerce vendors. Lost in the hype are the real security risks of e-commerce and these attributes make up only a small part of security, privacy and client trust of e-commerce risks. In what follows, we will discuss further about the advancement on security, privacy and client trust of e-commerce risks, making secure business application logic for e-commerce security, privacy and client trust in which we will describe about e-commerce security more than cryptography and the idea of secure business application logic for e-commerce.

Further advancements in the field E-commerce security; more than cryptography

As the market for e-commerce continues to heat up significantly, most stakeholders in the market are quick to declare that the system is secure. Encryption protocols such as SSL are promoted as proof of security by e-commerce vendors. Lost in the hype are the real security risks of e-commerce. Although encryption of data during transactions provides crucial confidentiality, integrity, and authentication, these attributes make up only a small portion of security that must be ensured for e-commerce security. Businesses engaging in the complex nature of business, e-businesses are much more vulnerable than simple websites due to the greater complexity of the software necessary to support e-commerce transactions (Garfinkel & Stafford, 1997; Gosh, 1998).The software that executes on either end of the transaction e sever-side or client-side software poses real threats to the security, privacy & client trust of all e-commerce transactions. Two familiar adages play an important role in understanding how to secure e-commerce systems: (1) a chain is only as strong as its weakest link; and (2) in the presence of obstacles, the path of least resistance is always the path of choice (Ghosh,2000). Encryption protocols generally provide the strongest perceived security in all the components involved in e-commerce transaction (Garfinkel and

Stafford, 1997).However, in e-commerce as in other real-world systems, the security of the system is only as strong as its weakest component. On the whole, the security of server-side systems is much weaker than the security provided by secure data transaction protocols such as SSL, example of this claim refers to the above mentioned cases of security breach and the case study of Barclay Bank Web site targeted that is the case of Copy Cat Trick which is conducted through the server-side misconfiguration of softwares, cause of vulnerability/compromising server-side security is clearly proof of this claim. Considering these two adages together in the context of e-commerce security, privacy & client trust, it becomes clear that malicious perpetrators will rarely attempt to break encryption codes when they can much more easily break into a system, and enjoy a much higher return via this simpler approach (Ghosh, 2000).This article addresses the topic of securing server-side software used in e-commerce applications. Today, most e-commerce server applications are implemented in n-tier architecture as shown in Fig. 1.This architecture usually consists of a front-end web server, a business application layer, a backend database, ERP systems, and legacy software. In practice, many of the security issues in e-commerce sites are specific to the way in which these components are configured.

A weak link: business application logic

Although the security issues of the front-end and back-end software systems in e-commerce applications warrant equal attention, this article focuses on the security of the middle tier of e-commerce servers that implements the business application logic. The business application logic represents the functions or services that a particular e-commerce site provides. As a result, a given site may often employ custom-developed logic. As the demand for e-commerce services grows, the sophistication of the business application logic grows accordingly. The end result is a complex custom software system that, if flawed, can result in a complete security compromise of the site (Ghosh, 2000).Traditionally, e-commerce sites implement the middle tier of software on web servers using the Common Gateway Interface (CGI). CGI scripts are programs that run on the web server machine as separate processes from the web server software.

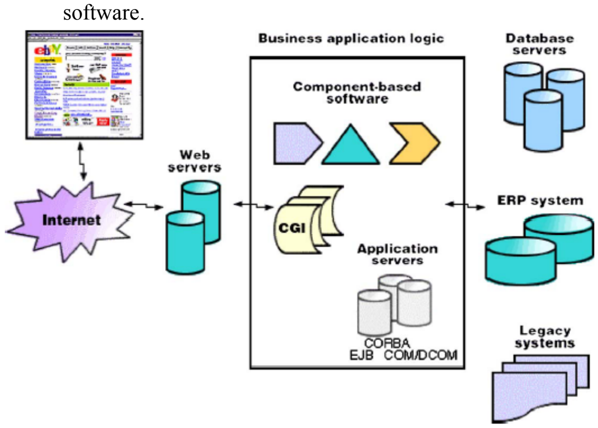


Figure 1 Secure business application logic model for e-commerce systems security.

The web server invokes these general-purpose programs in response to user requests. The CGI scripts' main function is to process user input and performs some services (such as retrieving data from a database, or dynamically creating a web page) for the user. Because CGI scripts process untrusted user input, the security risks associated with the CGI (and other forms of middle-tier software) are extremely high. Many attacks against web based systems exploit CGI scripts (Gundavaram, 1996).

Expecting the unexpected

While developers can write CGI scripts in any general-purpose programming language, they most often write such scripts in Perl, C, Tcl, or Python. The basic rule of thumb when designing CGI scripts is to expect the unexpected. Or more appropriately, expect the malicious. While developers have control over the content of CGI scripts, they have no control over what users will send to the CGI scripts. Also, do not overlook attacks against CGI scripts that exist on the server as part of the distribution, but are not even used as part of the e-commerce application. Some CGI scripts included as part of the web server distribution have well-known flaws that can be exploited to obtain unauthorized access to the server. Even if developers do not use the default CGI scripts as part of the web server pages, anyone else can use them by simply knowing the script names (Stein, 1998).More recently, component-based software (CBS) has made inroads in e-commerce applications. The idea of using CBS is to develop, purchase, and reuse industrial-strength software in order to rapidly prototype business application logic. One of the more popular component frameworks for e-commerce applications is Enterprise Java Beans (EJB), which supports component-based Java. Other component models include the Common Object Request Broker Architecture (CORBA), an open standard developed by the Object Management Group (OMG), and Microsoft's Common Object Model (COM) and Distributed COM (DCOM).The component frameworks are the ‘‘glue’’ that enables software components to provide services such as business application logic and use standard infrastructure services such as naming, persistence, introspection, and event-handling, while hiding the details of the implementation by using well-defined interfaces (Gosh, 1998).Fig. 1 shows typical n-tier architecture for a component-based e-commerce application. The web browser client can display web pages in HTML, execute mobile code such as Java applets or web scripts, and capture business-specific semantics using XML. The web server provides web services in addition to other Internet services such as e-mail and File Transfer Protocol (FTP). The business application logic is coded in software components that can be custom-developed or purchased off the- shelf. The application servers provide the infrastructural services for particular component models such as EJB, CORBA, COM, and DCOM. They also provide an interface for the business application logic to back-end services such as database management, enterprise resource planning (ERP),and legacy software system services.

Component-based software for business-to-business applications

In addition to supporting traditional CGI functions, component-based software is expected to enable distributed business-to-business applications over the Internet. The component-based software paradigm also supports good software engineering (see ‘‘Designing for Security’’ section). The Unified Modeling Language (UML) supports object-oriented analysis and design for component-based frameworks. In addition, as the market for component-based software heats up, many standard business application logic components will be available for purchase off the shelf.

The security risks of component-based software:

Although component-based software provides numerous benefits, it poses security hazards similar to CGI scripts. Component-based software enables software development in general-purpose programming languages such as Java, C, and C++. As such, these components execute with all the rights and privileges of server processes. Like CGI, they process untrusted user input. However, because component-based software can be used to build sophisticated large-scale applications, errors are unarguably more likely with component-based software than with simple CGI scripts. Regardless of the implementation CGI or application servers the security risks of server side software are high and therefore server-side software must be carefully designed and implemented using the techniques covered in the following sections (Stein, 1998).

Designing for security:

As with all software development, good design and engineering practices are important for software quality. This point is particularly important for development of security-critical software such as e-commerce applications. Rather than thinking of security as an add-on feature to software systems, security should be designed into the system from the earliest stages of requirements gathering through development, testing, integration, and deployment. The goal of design for security is to break the penetrate-and-patch mindset that pervades commercial software security today, and replace it with a process for finding and removing security-related bugs prior to software release. Finding and fixing bugs in software after release costs orders of magnitude more than correcting them early during the software development lifecycle (Gosh, 1998).One unfortunate consequence of the huge pressure to reduce time to market for e-commerce services is that good software engineering practices are dropped (or more often never started).As described in section ‘‘A key weak link: business application logic’’, the demand for e-commerce applications is driving the complexity of business application logic. Developing software such as a typical desktop application for one user at a time differs significantly from developing an e-commerce application that needs to handle tens of thousands of concurrent requests. Today’s e-commerce applications need to not only handle many simultaneous users, but also deal with malicious threats. As a result, developers must employ good software engineering practices to develop robust and secure business application logic. Several key activities mark good software engineering practices:

- Gathering and formally specifying requirements.
- Developing normal and pathological usage scenarios.
- Object-oriented analysis and design.
- Adopting good coding practices.
- Unit testing and integration.
- Release engineering.
- Third-party validation and verification.

Requirements’ gathering has traditionally focused on users’ needs for the application under development. In gathering requirements for security critical applications, developers must equally focus on what the user should be allowed to do and should not be allowed to do. a plan for meeting functional requirements is typically captured in specifications for the application. However, specifications for e-commerce applications need to specify not only which functional behaviour is expected from the application, but also which behaviour is not desired. Therefore, to develop a specification of secure behaviour, the project team must develop a security policy for the e-commerce application (and its users). Developing specifications is often fraught with anxiety over formality. While formal notations can be useful for reducing ambiguity, if they are so daunting that they discourage the development of specifications, then they can have the opposite effect on software quality. Remember that the ultimate goal of software specification is to promote understanding of intended system behaviour, so any specification is better than none.A specification of undesired behaviour is not only useful for preventing security design flaws, but can also be used for security-oriented testing (Gosh,1998).

Configuring the CGI:

One of the most commone yet easily preventable security hazards is misconfiguration of software.CGI scripts, too, must be correctly configured for security. One feature supported by many web servers is the ability for individuals throughout an organisation to write CGI scripts and have them executed from their own directories. Although useful for ‘‘sniffing’’ personal web pages, such scripts can introduce system security hazards. In e-commerce applications, the web server should be configured to prevent CGI scripts from executing anywhere but a single CGI directory controlled by the system administrator. The script-aliased CGI mode for web servers ensures that CGI scripts will execute only from an explicitly named directory in the server configuration file. In addition, the CGI script path is not named in the URL to the CGI.Rather, the server ‘‘aliases’’ the explicit path to the CGI scripts to a name of choice such as ‘‘cgi-bin’’Thus, running the server in script-aliased CGI mode prevents rogue CGI scripts from executing, and also hides the explicit path to the CGI scripts (Pre-Built CGI Scripts).Configure the CGI script directories correctly using operating system access controls. For instance, if CGI scripts are written in a compiled language (such as C), the script sources should be excluded from the document root of the web server, so that they cannot be accessed via the web. They should be accessible to the system administrator or web content development group only, and inaccessible to everyone else in the organisation. If the script sources fall into the hands of malicious perpetrators, the source code can be inspected for flaws, making the perpetrators’job even easier. Access to the CGI executables directory, frequently called the cgi-bin, should also be properly controlled. Only the web server and administrator need access to this directory. Liberal access permissions to the CGI executables directory let malicious insiders place their own scripts on the e-business site. Also, be sure to prohibit access to interpreters from the web server. For example, system administrators may be tempted to include the Perl interpreter in CGI script directories. However, doing so provides direct web access to interactively execute Perl commands e an extremely dangerous configuration. Finally, account for every CGI program on the server in terms of its purpose, origin, and modifications. Remove CGI scripts that do not serve a business function. View with suspicion and carefully screen CGI scripts that are distributed with web servers, downloaded from the Internet, or purchased from third parties. These steps will eliminate most of the potentially dangerous CGI scripts. Once a stable set of CGI programs has been established, make a digital hash of the executable programs (using MD5) to enable future integrity checks (Christiansen and Torrington, 1998).

Defensive programming

Simply configuring the system securely is not enough to ensure security. Even after the CGI scripts are properly locked down, flaws in the design and implementation of the CGI scripts can be exploited to leverage full system compromise.A good starting place for writing secure CGI scripts is to use safe language features. For instance,although interpreted languages such as Perl are great for rapidly prototyping software, they also provide powerful constructs for executing system commands that perpetrators can easily exploit.Perl commands such as eval (), system (), backquotes (‘), pipes, and exec ( ) can potentially result in system commands being executed on the web server at the discretion of an unknown and untrusted remote user. Use these commands with extreme caution when processing user input. Although the same types of system commands can be used in compiled languages such as C, they are not as easily constructed or misused as system commands in interpreted languages such as Perl,Python, or Tcl. However, C provides its own hazards due to its lack of advanced memory management. For instance, the most common and notorious security-related bug for C programs is the buffer overflow. This condition occurs when the memory allocated to a variable is less than the amount being written into it. Regardless of the language used for implementing the business application logic, the Cardinal’s rule in development is never to trust the user input.

Sanitising user input

Fortunately, defensive programming can add security where powerful language constructs might otherwise fall short. A key defensive strategy is to sanity-check (and sanitise) all program input. For instance, if a social security number is requested, be sure that the input is numeric and that no more than nine characters are read. Also be sure to sanitise input of special Meta escape characters. Characters such as the backtick (‘) allow input to be executed as commands to the interpreter.Unescape special characters, or if possible, accept only legitimate characters. Do not assume that the length of the input is fixed. Even if the length of the input field is limited in a form, the user is not constrained to using the form to send in user input. When parsing user input, read only the length of data necessary for the input and ensure that adequate memory is allocated for the amount of input to be read (Ghosh, 2000).

Using environment variables

A common oversight in CGI script development is the use of environment variables. The user can set certain environment variables. Sanitise or sanitycheck environment variables from HTTP clients before using them. Use other system environment variables with caution. Specify explicit paths rather than rely on system environment variables for accessing programs and files. An insider can fool a CGI script that depends on an environment variable to execute a rogue program instead of the intended program. Also, do not use hidden fields in forms in order to hide information. Though hidden fields may not display in the browser, they are easily discerned by inspecting the HTML source.

Leveraging existing safety features

In addition to these defensive programming techniques, some languages provide safety features that should be leveraged when possible. For instance, Perl 5 contains a ‘‘taint’’ module for preventing user-supplied input from being used in system commands. When Perl tainting is enabled, any variable derived from user input (such as command-line arguments and environment variables)is considered tainted (or untrusted). Thus, when a tainted variable is used in a system command, such as modifying the file system or spawning a shell, the Perl runtime interpreter prevents the command from executing. Similarly, Tcl has a safe language interpreter that can prevent untrusted input from compromising system security and integrity. When using application servers for implementing business application logic, prefer using strongly typed languages such as Java rather than C or C++. Java’s type safety and advanced memory management can eliminate typical programmer errors in walking through memory that often lead to security breakdowns. Furthermore, Java 2 provides fine-grained access control for Java programs to prevent unauthorized access to system resources. In summary, always expect the unexpected from user input. Employ defensive programming techniques such as sanitising input regardless of the language used. If you develop in an interpreted language, then use the safe language-interpreter features such as Perl tainting and Safe-Tcl. When developing business application logic from components, use a type-safe language such as Java rather than languages such as C and C++, which give free reign to the programmer to use memory objects in any context.

Wrapping the business application logic

The defensive programming techniques described in the previous section can significantly help to write secure server-side software. Even so, perfection is largely unattainable in software development. To improve the security of the business application logic, consider using one of several existing technologies that can ‘‘wrap’’server-side software in order to limit potential damage from flawed software exploited for malicious gain. One of the most commonly employed approaches on UNIX-based systems is to use the chroot () command to effectively cordon off the file system visible to a program. The chroot () function changes the effective root of the file system for a given process to a designated new root. For server-side software, define the new root as the smallest partition of the file system necessary for the programs to access. When a process is run in a chroot () environment, it is effectively running in a jail cell to prevent it from causing damage to other portions of the system. While the chroot () environment is useful for preventing auxiliary damage to the rest of the system, it is important to remember that the chroot () environment must contain the portion of the file system necessary for the program to perform its function. If the business application logic needs to communicate with critical system resources such as back-end databases, then these resources need to be included within the chroot () environment. In this case, the chroot () environment will not protect databases or other included resources from misbehaving software, but will protect the integrity of other system resources outside the chroot () environment. Alternative commercial technologies for ‘‘sandboxing’’ executing processes are available on the Windows platform. Another technique for UNIX-based systems provides some protection by running CGI scripts with user permissions. The program CGIWrap is designed for systems that allow internal users to write and post their own CGI scripts. Although allowing users to post their own CGI scripts is not advisable for e-commerce applications, if it is a requirement, then CGIWrap can be useful for limiting the privilege of the executing CGI process to that of the user. The technique assumes that permissions of users are properly configured on the system such that no user is allowed access to other users' files or to critical system files and programs. In many systems, the web server is configured to run all CGI scripts under a single account's privileges. Usually an account is created for this purpose, such as www or nobody. However, in systems where internal users are allowed to post their own CGI scripts, it is not advisable to allow a user's script to access files owned by other users' or other web server files. CGIWrap enforces the policy that users' CGI scripts can only access files owned by the script's owner. Again, before leaving the discussion of CGIWrap, it is important to reiterate that it is generally not advisable to allow users to post their own CGI scripts to an ecommerce site.A more attractive alternative to CGIWrap is the SBOX tool written by Stein (1998). Like CGIWrap, SBOX is designed for multi-user websites where users are allowed to post their own CGI scripts. However, in addition to limiting the privilege of CGI scripts to their owner's files, SBOX can also place limits on executing CGI processes on the use of system resources such as CPU cycles, memory, and disk. This feature provides an effective way to the art denial-of-service attacks from malicious CGI scripts, or poorly written CGI scripts that end up consuming system resources to the detriment of the other e-commerce services. SBOX also allows CGI scripts to be run in a chroot () directory. Thus, SBOX provides wrapping capability of programs that not only limits the file system using chroot (), but also ensures that the CGI scripts cannot infringe on other users' programs and will not deny service to other executing processes. It is important to stress again; however, that e-commerce applications should not be deployed in a general-purpose environment that allows multiple their CGI scripts. But if CGI uploading does need to be supported, tools such as SBOX and CGIWrap can reduce the likelihood of catastrophic security breaches.

Conclusion

E-commerce security, privacy and client trust issues should always be dealt with when surfing on the web or using the Internet for business purposes.E-commerce encompasses all aspects of using the Internet for business or personal use.Now, more than ever, a great deal of business is performed in one-way or another over the Internet. For some, it is simply ease of communication; for others, having the ability to research topics, products, or even people make the Internet an absolute necessity for business. Businesses have begun exploiting the Internet for commercial transactions. Recognizing the dangers in sending confidential information over an inherently insecure media, a number of secure data transport protocols have emerged. At minimum, these protocols encrypt sensitive information such as credit card numbers to prevent unauthorized people from capturing the data. Some protocols even facilitate payment for merchants through banking institutions. Even with the strong security provided in the transport of data, e-commerce security still remains elusive. In practice, most security violations occur through other avenues than breaking cipher text. Using encryption on the Internet is the equivalent of arranging an armoured car to deliver credit-card information from someone living in a cardboard box to someone living on a park bench.The point is that usually we infer security from encryption when we are so vulnerable otherwise. Therefore, keeping in view of strategy of developing secure business application logic for ecommerce sites (server-side security and privacy)can secure from breaching by securing server-side software, because security is as strong as its weakest link. This is why common sense is the best tool to make your side secure from hacking/breaching system. Common sense is the appropriate tool that can be used to establish your security policy. Elaborate security schemes and mechanisms are impressive and they do have their own reasons, but yet there is little point in investing money and time on an elaborate implementation scheme if the simple controls are forgotten.

References

Ahuja V. E-commerce on Internet. London: Academic Press Ltd;1997.

Chaffy D. Business information systems; technology, development and management in the e-business; 2000. ISBN:027365540X.

Christiansen Tom, Torkington Nathan. Perl cookbook: O'Reilly & Associates; 1998. [chapter 19: CGI programming].

Ernst, Young. Internet shopping e an Ernst & Young special report; 1998 [Section 2].

Foresight. E-commerce sets new rules, Systems Relationships Marketing, on behalf of Datatec Ltd, 1998 November;1(3).

Furche A, Wrightson G. Asystematic overview of electronic payment system. Heidelberg, FDR: dpunkt, Verlag, Fur digtiale Technology Gmbh; 1996.

Garfinkel Simson, Stafford Gene. Web security and commerce:O'Reilly Publishing; 1997.

Gundavaram Shishir. CGI programming on the World Wide Web:O'Reilly & Associates; 1996.

Ghosh Anup K. director of security research at software technologies. Security and privacy in e-commerce. John Wiley & Sons; 2000.

Gosh Anup K. E-commerce security: weak links, best defences:John Wiley & Sons; 1998. ISBN: 0-471-19223-6.

Hunt Craig. TCP/IP for Unix administrators. 2nd ed. O'Reilly;1998. ISBN: 1-56592-322-7.

Kalakota Ravi, Whinston Patrick. Frontiers of electronic commerce: Addison-Wesley; 1996.

Kamthan Pankaj. A matter of trust. Journal of E-Commerce on the WWW Sunday 25th April 1999.

Labuschagne L, Eloff JHP. E-commerce: the information security challenges journal. Information Management & Computer Security 2000;8(3):154e7 ISBN: 0968-5227. MCB University Press.

Lawrence Elaine, Corbett Briam. E-commerce and Internet digital models for business. 2nd ed. Australia: John Wiley & Sons; 2000. ISBN: 0 471673.

Pre-Built CGI Scripts Security Issues When Installing and Customizing Pre-Built Web Scripts, Selena Sol, WDVL.com.

Sielgel C. Internet security for business. New York: John Wiley & Sons, Inc; 1996.

Stein Lincoin. Web security step-by-step reference guide.Addison Wesley; 1998.

A Scale-based Reduction Method for Surveillance of Massive Moving Objects on Mobile Devices

Hsien-Chou Liao and Wen-Feng Chen
Department of Computer Science and Information Engineering
Chaoyang University of Technology, Taiwan
E-mail: hcliao@mail.cyut.edu.tw; fwen@cht.com.tw

Abstract

Recently, the wireless network and mobile devices is getting popular. It causes that many mobile information services are implemented on such devices. The implementation is concerned with four main resources: storage, computing speed, screen size, and network bandwidth. In this paper, a mobile client for surveillance of massive mobile objects is considered. The locations of objects are retrieved from a server and updated periodically. Such application is suffered from the constraints on screen size and network bandwidth. A scale-based reduction method (SRM) is proposed to overcome the above constraints. Scale is an instinctive way to solve the constraints on screen size. Locations are reduced based on the scale and thus decrease the utilization of network bandwidth. Experiments are designed to evaluate the reduction ratio and transmission ratio of SRM. The results illustrate that SRM is effective for surveillance of massive moving objects on mobile devices.

Keywords: location reduction, surveillance, mobile application

1. Introduction

The wide spread of wireless network and mobile devices causes that the mobile information services are getting popular. However, the development of such services is mainly constraints by four resources: storage, computing speed, screen size, and network bandwidth. For example, current Pocket PC is equipped with 64MB or 128MB SDRAM, 400MHz CPU. Its screen size is 240×320 pixels. The bandwidth of 802.11g wireless network is 54Mbps. Many services need special design to meet the above constraints.

On the other hand, the locations of moving objects can be acquired from positioning devices. These devices can be classified into two categories: outdoor or indoor. Global positioning system (GPS) is the most popular outdoor positioning device. However, it cannot be used in the indoor environment. Many indoor positioning technologies are proposed. They rely on various kinds of devices, such as RFID, ultra wideband, wireless LAN, or even Bluetooth [1-4]. In addition, many researches are focused on location-related topics. For example, S. Banerjee *et al.* proposed a location-aware system architecture, called Rover [5]. It is designed to scale to large user populations. When a user moves from the service domain of a Rover into another, the user's profile, e.g., preference, device capability, *etc*., can be transmitted to the target Rover. Therefore, the target Rover can provide the service without interruption. Some researches are focused on the location management. For example, S. C. Lo *et al.* propose a method to solve the performance problems on location management for personal communications services (PCSs) [6]. In the proposed method, the user's movement behavior is associated with a set of regions. The registrations process in the same region can be eliminated such that the cost of location management can be significantly reduced. D. L. Lee *et al.* clarifies many issues related to the location-dependent information service (LDIS) [7]. These issues include location-dependent data placement, replication, indexing, caching, broadcasting, query scheduling, *etc*. Besides, M. Bauer *et al.* focus on the location model and derive requirements for a general location modeling language for ubiquitous computing, Augmented World Modeling Language (AWML) and Augmented World Querying Language (AWQL)[8]. Many researches also study how to identify the location of a user precisely based on the movement behavior, the cell-ID in GSM cellular network, or signal strength in wireless network [9-11]. S. Harri *et al.* focus on the location-dependent query, e.g., "Please list the taxi within five miles?" Since the query results need be updated continuously, agent technology is used to improve the update efficiency [12]. S. Prabhakar *et al.* propose another solution to overcome the same problem [13].

The popular of mobile devices and the trend of location-aware computing causes that more and more location-related services are implemented on such devices. One of the basic services is the surveillance of moving objects. If the objects are massive and the area is large, the service is constrained by the screen size and network bandwidth. In this paper, a scale-based reduction method (SRM) is proposed to overcome the above constraints. Scale is an instinctive way to solve the constraints on screen size. The locations can be reduced according to the scale determined by the user. The visible area is also dependent on the scale. Only those locations in the visible area are transmitted to the mobile devices. Therefore, the network transmission is reduced, too. In our experiment, two ratios, reduction ratio and transmission ratio, are defined to evaluate the performance of SRM. The experimental results show that SRM is effective for surveillance of massive moving objects on mobile devices.

In the following materials, the reduction principle of SRM is presented in next section. The simulation tool is presented in Section three. The experimental result is presented in Section four. Section five is the conclusion and future works.

2. Method

Assume the locations of moving objects can be acquired from carried devices periodically, such as Pocket PC, smart phone, palm, *etc.*; the locations are then transmitted and stored in a location server. A surveillance service on mobile devices is mainly used to display instant locations of objects. Such a scenario is shown in Fig. 1:

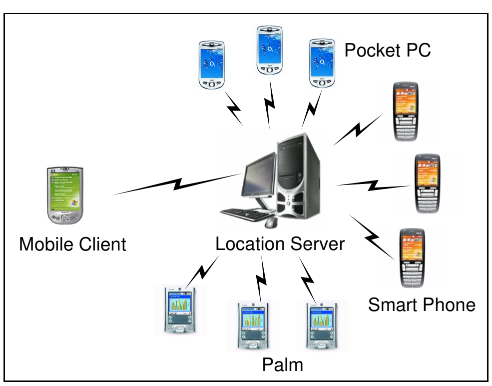


Fig. 1. A surveillance service scenario

Before describing the principle of SRM, the distinction between data reduction and compression methods are discussed first. In general, data compression method is popular used to decrease the data size. In addition, data is compressed on the server side and decompressed on the client side. Data is usually compressed as a whole. The compression methods may achieve small compression ratios depended on the type of data. Oppositely, the data reduction method proposed in this paper is only executed on the server side. It can reduce partial of the data depending on user's need. The reduction ratio is usually larger than the compression ratio. The comparison of data compression and data reduction are listed in Table 1:

Table 1. The comparison of data reduction and data compression

Methods / Factors	Data Compression	Data Reduction
Load	Server/Client	Server
Scope	All	Partial
Ratio	Small	Large

If the compression method is used in the scenario of Fig. 1, the locations must be decompressed periodically in order to get the instant locations. The method suffers from the constraint on computing speed of mobile devices. Besides, the device's screen size is small. It is unnecessary to compress and transmit all of the locations to the mobile client. The screen size causes that users can only focus on a part of the area. Only partial of the locations is transmitted. Consequently, reduction method is suitable for the above scenario.

The principle of SRM is illustrated by the example in Fig. 2:

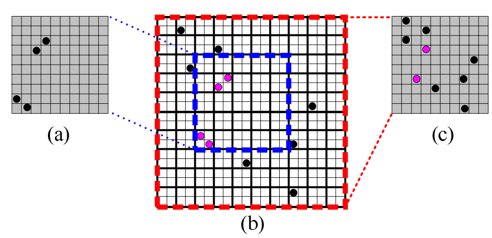


Fig. 2. The illustration of SRM principle (a) scale: 1/1 (b) original area (c) scale: 1/2

Assume the surveillance area is 20 meters×20 meters and the screen size is 10 pixels×10 pixels. The original area is shown in Fig. 2(b). The black points represent the instant locations of objects. In Fig. 2(a), the scale is 1/1 that means one pixel represents one meter. Only one fourth of the area is visible. One of the visible areas is marked with blue dash line. Only four locations in the area are transmitted to the mobile client. In the Fig. 2(c), the scale is 1/2 that means one pixel represents four square meters. The original area with red dash line is visible on the screen. However, the location is imprecise under the scale 1/2. The actual location of one pixel could be one of the 2×2 square meters. If two locations are within the same square, they are reduced to one pixel in the screen. In the above figures, four pink points in Fig. 2(b) are reduced to two pink points in Fig. 2(c). That is, even though the whole area is displayed on the device, the transmission size is usually smaller than the original size.

According to the above principle, users can adjust the scale and focus on the interesting area, i.e., visible area. Only the locations in the area are reduced and transmitted to the mobile client. There is no computing load on the mobile device.

In order to evaluate the performance of SRM, two ratios, reduction ratio (*RRatio*) and transmission ratio (*TRatio*), are defined. *RRatio* is the percentage of total data bytes after reduction and *TRatio* is the percentage of data bytes to be transmitted.

In practice, maps or labels of objects should be transmitted to the mobile clients for surveillance purpose. However, these data are fixed and can be transmitted at the starting of the surveillance service. They are not included in the computation of *RRatio* and *TRatio*.

Assume the original area is a square and its width is denoted as W. The screen width is denoted as SW. The scale is denoted as S. S equals to one if the scale is 1/1 and equals to 0.5 if the scale is 1/2. The number of locations is LS. L_i represents one of the locations, and (x_i, y_i) represents the coordinate of L_i. The number of bytes for storing a coordinate, N, can be reduced when scaling is used. The number of bytes for storing the identification of a moving object is denoted as M. After the reduction of SRM under the scale S, the coordinate of L_i is transformed to $L_i^{'}$ where the equation is defined as follows:

$$L_i^{'} = \left(x_i^{'}, y_i^{'}\right) \text{where } x_i^{'} = \lfloor x_i \times S \rfloor, y_i^{'} = \lfloor y_i \times S \rfloor \quad (1)$$

The coordinate of $L_i^{'}$ equals to the original coordinate multiplies the scale S. Before computing the *RRatio*, the total data bytes must be computed. The equation is defined as follows:

$$Count_{mn} = \text{Number of } L_i^{'} \text{ where } x_i^{'} = m \text{ and } y_i^{'} = n$$
$$B_{mn} = \begin{cases} N + M \times Count_{mn} \text{ when } Count_{mn} \geq 1 \\ 0, \text{otherwise} \end{cases} \quad (2)$$
$$Total_S = \sum_{m=1}^{\lfloor W \times S \rfloor} \sum_{n=1}^{\lfloor W \times S \rfloor} B_{mn}$$

In Equation (2), the number of locations $L_i^{'}$ that equals to the coordinate (m, n) of the reduced area is denoted as $Count_{mn}$. B_{mn} represents the data bytes with respect to the coordinate (m, n). When $Count_{mn}$ is greater or equals to one, $M \times Count_{mn}$ represents the total bytes for the identifications of $Count_{mn}$ objects. N represents the bytes for the coordinate (m, n). Therefore, B_{mn} equals to the $N + M \times Count_{mn}$. Oppositely, it equals to zero when $Count_{mn}$ is zero. The summation of B_{mn}, i.e., $Total_S$, represents the total data bytes under the scale S.

Then, $RRatio_S$ represents the reduction ratio under the scale S. It equals to $Total_S$ divided by the total data bytes of the original area, i.e., $Total_1$. It is defined as follows:

$$RRatio_S = \frac{Total_S}{Total_1} = \frac{Total_S}{(N + M) \times LS} \quad (3)$$

In advance, the visible area influences the transmission size. For example, the size of a crowned area must be larger than that of a thin area. Therefore, three kinds of *TRatio* are defined to realize the range of transmission ratio. They are *MaxTRatio*, *MinTRatio*, and *AvgTRatio*. Their equations are defined as follows:

$$\begin{aligned} TRatio_S(m,n) &= \frac{\sum_{i=m}^{m+SW-1} \sum_{j=n}^{n+SW-1} B_{ij}}{Total_1} \\ MaxTRatio_S &= Maximum\, TRatio_S(m,n), \\ &\quad 1 \leq m \leq \lfloor W \times S \rfloor - SW + 1, \\ &\quad 1 \leq n \leq \lfloor W \times S \rfloor - SW + 1 \\ MinTRatio_S &= Minimum\, TRatio_S(m,n), \\ &\quad 1 \leq m \leq \lfloor W \times S \rfloor - SW + 1, \\ &\quad 1 \leq n \leq \lfloor W \times S \rfloor - SW + 1 \\ AvgTRatio_S &= \frac{\sum_{i=1}^{\lfloor W \times S \rfloor - SW + 1} \sum_{j=1}^{\lfloor W \times S \rfloor - SW + 1} TRatio_S(i,j)}{\left(\lfloor W \times S \rfloor - SW + 1\right)^2} \end{aligned} \quad (4)$$

In Equation (4), $TRatio_S$ (m, n) is the transmission ratio of a visible area where the coordinate of its left-upper corner is (m, n). It equals the summation of B_{ij} in the area divided by the $Total_1$. $MaxTRatio_S$ and $MinTRatio_S$ equal to the maximum and minimum value among $TRatio_S(m, n)$, respectively. The $AvgTRatio_S$ is the average of $TRatio_S(m, n)$.

Using Fig. 1 as an example, the area width W equals to 20, the screen width SW equals to 10, and the scale equals to one or 0.5. The value of N is assigned as two bytes and M is assigned as two bytes, too. The number of locations, LS, equals to 11. The computations of *RRatio* and *TRatio* are shown as follows:

$$RRatio_{0.5} = \frac{Total_{0.5}}{Total_1} = \frac{(2+2)\times 7+(2+2\times 2)\times 2}{(2+2)\times 11}$$
$$= 90.9\%$$
$$RRatio_1 = \frac{Total_1}{Total_1} = \frac{(2+2)\times 11}{(2+2)\times 11} = 100\%$$
$$TRatio_{0.5}(1,1) = \frac{(2+2)\times 7+(2+2\times 2)\times 2}{(2+2)\times 11} = 90.9\%$$
$$MaxTRatio_{0.5} = MinTRatio_{0.5} = AvgTRatio_{0.5} = 90.9\%$$
$$TRatio_1(1,1) = \frac{20}{44} = 45.5\% \qquad (5)$$
$$TRatio_1(1,2) = \frac{20}{44} = 45.5\%$$
$$\ldots$$
$$MaxTRatio_1 = \frac{20}{44} = 45.5\%$$
$$MinTRatio_1 = \frac{4}{44} = 9.1\%$$
$$AvgTRatio_1 = \frac{36.91}{(11)^2} = 30.5\%$$

When S equals to 0.5, the total data bytes are reduced to 90.9%. Only 9% is reduced by SRM. Since all the data must be transmitted to the mobile client, the $MaxTRatio_{0.5}$ also equals to 90.9%. On the other hand, when S equals to one, $RRatio_1$ is 100% and $AvgTRatio_1$ is 30.5%. It means that even although no reduction is achieved under the scale one, only one third of the data bytes are transmitted in average.

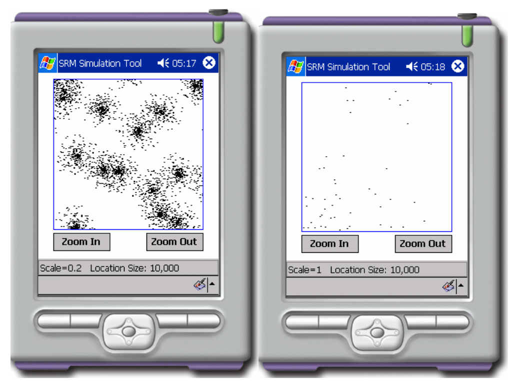
3. Simulation Tool

A simulation tool is implemented by using VB.NET for evaluating the performance of SRM. The default area is 2,000 meters × 2,000 meters. The screen size is 200 pixels × 200 pixels. The location size can be 1000, 4000, 7000, and 10000. They are generated based on random or group distributions. Under the group distribution, the default group size is 40 and 250 locations for each group. The area of a group is a circle with 200 meters radius. The simulation tool is illustrated by the screen shots in Fig. 3.

When the scale is 0.1, the original area is visible on the screen as shown in Fig. 3. These locations are generated based on group distribution. Users can press the "Zoom In" or "Zoom Out" buttons to adjust the scale. When the scale is adjusted to 0.2, the screen shot is shown in Fig. 4(a). In Fig. 4(b), the scale is changed to one. The locations can be shown on the screen precisely. In Fig. 5, the screen shot shows locations generated based on the random distribution.

According to the above illustrations, we can understand easily how a tool based on the SRM is worked.

Fig. 3. The screen shot of a group distribution with scale 0.1 (1/10)



(a) (b)

Fig. 4. The screen shots of a group distribution with different scales (a) 0.2 (1/5) (b) one (1/1)

Fig. 5. The screen shot of a random distribution with scale 0.1 (1/10) and 4,000 locations

4. Experimental Results

The purpose of the experiment is to evaluate the performance of SRM. The parameters are listed below:

- Original area: 2,000 meters × 2,000 meters
- Screen width: 200 pixels × 200 pixels
- The number of locations: 1000, 4000, 7000, and 10000
- Scale: 0.1, 0.2,…, and 1.
- Location distribution: random or group. For group distribution, there are 250 locations for each group.
- *N*: two bytes for storing a coordinate on mobile devices
- *M*: two bytes for storing the identification of a moving object

The screen width is 200. One byte is enough for storing the x or y-coordinate. Therefore, *N* is assigned as two bytes. *M* is two bytes which means there are maximal 65,536 moving objects can be identified.

Ten sets of locations are genereated separately. The *RRatio* and *TRatio* under different scales are computed for each set of locations. The results of *RRatio* are shown in Fig. 6.

In Fig. 6, the x-axis represents the scale from 0.1 to one. The y-axis represents the reduction ratio. There are four curves representing the *RRatio* of 1000, 4000, 7000, and 10000 locations with group and random distributions, respectively. In general, *RRatio* is increased as the increase of the scale. The *RRatio* of group distribution is about ten to fifteen percent lower than that of random distribution. It means that SRM method has better performance when the locations are concentrated in a small area. Four curves of random distribution are almost the same. It means that *RRatio* is not influenced by the location size under random distribution. Oppositely, the more the location is, the lower the *RRatio* can be achieved under the group distribution. The smallest *RRatio* is 80.6% under the scale 0.1 and 10,000 locations. Almost one fifth of location data can be reduced.

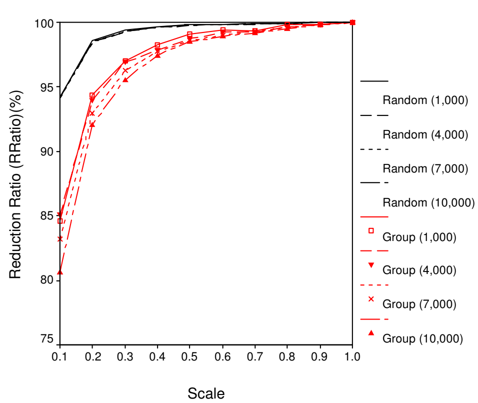


Fig. 6. The comparison of *RRatio* under random or group distributions

The results of *TRatio* are shown in Fig. 7. There are three curves representing the average, maximum, and minimum *TRatio*s under random and group distribution with 10,000 locations, respectively. It is obvious that *TRatio* is decreased rapidly as the increase of the scale. For the random distribution, the difference between *MinTRatio* and *MaxTRatio* is small. For the group distribution, *AvgTRatio* is close to the curve of random distribution. However, the *MaxTRatio* is about five to thirteen percent higher than *AvgTRatio*. It means that even though the locations are concentrated on a hot spot, *TRatio* is not increased too much. The smallest *MaxTRatio*, *AvgTRatio*, and *MinTRatio* are 6.2, 1.2, and zero percent, respectively. It illustrates that *TRatio* can be decreased rapidly by SRM based on the scale concept.

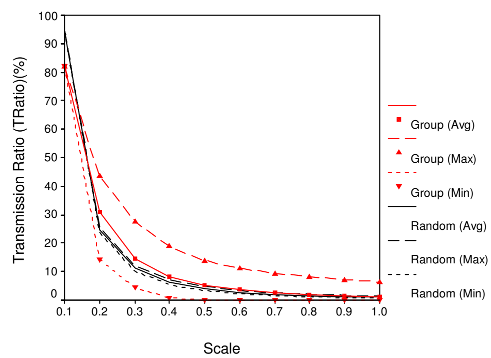


Fig. 7. The comparison of *TRatio* with 10,000 locations under random or group distributions

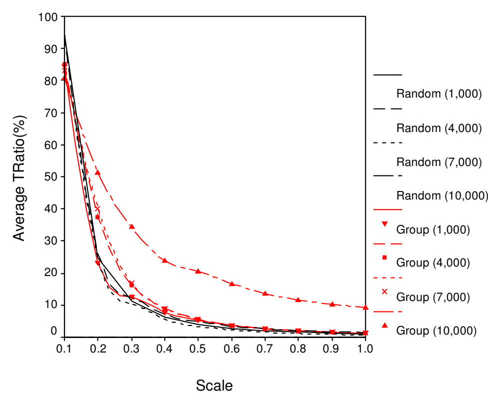


Fig. 8. The comparison of *AvgTRatio* with different location sizes

Besides, the influence of location size on the *AvgTRatio* is also experimented. The results are shown in Fig. 8. There is no explicit difference among the curves of random distribution. Oppositely, *AvgTRatio* is decreased as the decrease of the location size for the group distribution. It is because that the decrease of location size causes the group distribution is similar to random distribution. Therefore, the curve of group distribution with 1000 locations is overlap with the curve of random distribution.

According to the above results, although the smallest *RRatio* is 81 percent, *AvgTRatio* can be decreased to one and 1.2 percent under the random and group distributions, respectively. It illustrates that SRM can overcome the constraints on screen size and network bandwidth, and is effective for the surveillance of massive objects on mobile devices.

5. Conclusion and Future Works

The trend of implementing location-related services on mobile device is increasing. In order to implement a surveillance service of massive moving objects on mobile device, the constraints on screen size and network bandwidth are encountered. In this paper, a scale-based reduction method (SRM) is proposed to overcome the constraints. The experimental results illustrate that the SRM can reduce the transmission size and overcome the screen size by utilizing scale concept.

Some issues will be addressed in the future. For example, different users may subscribe various sets of moving objects for surveillance. These sets of locations must be reduced separately. SRM should be refined to provide the same service under various subscriptions. In addition, various subscriptions causes serious computing load on the server. It is also an important issue to be addressed to provide an ultimate location surveillance service.

Acknowledgements

This work was funded by the National Science Council under grant NSC93-2213-E-324-028.

References

[1] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, “LANDMARC: Indoor Location Sensing Using Active RFID,” Proc. of the First IEEE Int’l Conf. on Pervasive Comp. and Comm. (PerCom’03), March 23-26, 2003, pp. 407-415.

[2] T. Kitasuka, T. Nakanishi, and A. Fukuda, “Wireless LAN Based Indoor Positioning System WiPS and Its Simulation,” Proc. of IEEE Int’l Pacific Rim Conf. on Comm., Comp., and Signal Processing (PACRIM’2003), Vol. 1, 28-30 Aug., 2003, pp. 272-275.

[3] S. J. Ingram, D. Harmer, and M. Quinlan, “Ultra Wideband Indoor Positioning Systems and Their Use in Emergencies,” Proc. of IEEE Position Location and Navigation Symposium (PLANS’2004), 26-29 April 2004, pp. 706-715.

[4] G. Anastasi, R. Bandelloni, *et al*., “Experimenting an Indoor Bluetooth-Based Positioning Service,” Proc. of the 23rd IEEE Int’l Conf. On Distributed Computing Systems, 19-22 May 2003, pp. 480-483.

[5] S. Banerjee et al., “Rover: Scalable Location-Aware Computing,” IEEE Computer, Vol. 35 Issue 10, Oct. 2002, pp. 46-53.

[6] S. C. Lo and L. P. Chen, “Adaptive Region-Based Location Management for PCS System,” IEEE Trans. On Vehicular Technology, Vol. 51, No. 4, July 2002, pp. 667-676.

[7] D. L. Lee, J. Xu, and B. Zheng, “Data Management in Location–Dependent Information Services,” IEEE Pervasive Computing, Vol. 1, Issue 3, July-Sep. 2002, pp. 65-72.

[8] M. Bauer, C. Becker, and K. Rothermel, “Location Models from the Perspective of Context-Aware Applications and Mobile Ad Hoc Networks,” ACM Personal and Ubiquitous Computing, Vol. 6, Issue 5-6, Jan. 2002, pp. 89-94.

[9] M. H. Jin, Eric H. K. Wu, and J. T. Horng, “Location Query Based on Moving Behavior,” Proc. of the Eleventh IEEE Int’l Conf. on Comp. Comm. and Networks, 14-16 Oct. 2002, pp. 268-273.

[10] T. Roos, P. Myllymaki, and H. Tirri, “A Statistical Modeling Approach to Location Estimation,” IEEE Trans. on Mobile Computing, Vol. 1, Issue 1, Jan.-March 2002, pp. 59-69.

[11] D. Pandya, R. Jain, E. Lupu, “Indoor Location Estimation Using Multiple Wireless Technologies,” The 14th IEEE Proceedings on Person, Indoor and Mobile Radio Comm. (PIMRC’2003), Vol. 3, 7-10 Sept. 2003, pp. 2208-2212.

[12] S. Harri, E. Mena, and A. Illarramendi, “Dealing with Continuous Location-Dependent Queries: Just-in-Time Data Refreshment,” Proc. of the First IEEE Int’l Conf. on Pervasive Computing and Comm., 23-26 March 2003, pp. 279-286.

[13] S. Prabhakar, Y. Xia, et al., “Query Indexing and Velocity Constrained Indexing: Scalable Techniques for Continuous Queries on Moving Objects,” IEEE Trans. on Computers, Vol. 51, No. 10, Oct. 2002, pp.1124-1140.

Attack and Improvement on the Lee-Chiu Remote Authentication Scheme

Yung-Cheng Lee[1] and Wen-Chung Kuo[2]

[1]Department of Electrical Engineering, WuFeng Institute of Technology, Chaiyi, Taiwan.

[2]Department of Information Engineering and Computer Science, National Formosa University, Yunlin, Taiwan.

Abstract

Remote password authentication schemes provide users with secure mechanisms to access authorized data. Recently, Lee and Chiu proposed a remote authentication scheme with smart card; their scheme can resist forgery attacks and users can freely choose and change their passwords. However, in this paper, we shall show that Lee and Chiu's scheme has a security flaw that an adversary can guess the password and impersonate the legal user to login to the system. Finally, we propose an improvement scheme to fix the flaw.

Keywords: Cryptography; Password authentication; Guessing attack; Smart card.

1. Introduction

Remote password authentication schemes are the most widely used mechanism to access authorized data through networks. Until now, many schemes have been proposed to provide secure and efficient methods for remote authentication. Among them, some use smart cards to enhance the security [1–4], and some employ one-way hash functions to reduce the computation complexity.

Lamport [5] proposed a remote authentication scheme for insecure communication in 1981. In that scheme, the remote server needs to maintain a password verification table. In 2003, Wu and Chieu [6] proposed a remote authentication scheme with smart card; their scheme provides users to freely choose and change their passwords. Recently, Lee and Chiu pointed out that the Wu and Chieu's scheme is vulnerable to the forgery attack [7]. Lee and Chiu have also proposed an improvement scheme (denoted as *the Lee-Chiu scheme* hereafter) to mend this security flaw. In this paper, we prove that the Lee-Chiu scheme is still insecure in the way that an adversary can guess the password off-line and impersonate the legal user to login to the system. Finally, we propose an improvement scheme to resist the attack.

This paper is organized as follows: The Lee-Chiu scheme is reviewed in Section II; the attack on their scheme is shown in Section III. Then, an improvement scheme is proposed in Section IV. Finally, we make a conclusion.

2. Review of the Lee-Chiu scheme

The Lee-Chiu scheme has three phases: the registration phase, the login phase and the authentication phase. Their scheme is briefly described as follows.

2.1 The registration phase

In the registration phase, the user U_i first sends his/her identifier ID_i and password PW_i to the remote server in person or through a secure channel. Upon receiving the registration request, the server takes the following steps:

(1) Compute $A_i = h(ID_i, x)$, where x is the secret key of the server and $h(\cdot)$ is a collision resistant one-way hash function.

(2) Compute $B_i = g^{A_i h(PW_i)} \bmod p$, where p is a large prime number, and g is a primitive element in $GF(p)$.

(3) Install the information $\{ID_i, A_i, B_i, h(\cdot), p, g\}$ into a smart card and forward it to the user U_i.

2.2 The login phase

In the login phase, U_i attaches his/her smart card to the card-reader of a terminal and inputs his/her identity ID_i and password PW_i*. The smart card performs the following steps:

(1) Compute $B_i* = g^{A_i h(PW_i*)} \bmod p$.

(2) Compare B_i and B_i*. If they are identical, go to the next step; otherwise, reject the login request.

(3) Compute $Z_i = B_i \times A_i \bmod p$ and $C1 = h(B_i \oplus T)$, where T is the current time.

(4) Send the message $m = \{ID_i, Z_i, C1, T\}$ to the remote server.

2.3 The authentication phase

If the remote server receives the login request m at time T', the server performs the following steps to authenticate the user.

(1) Check ID_i and T. ID_i must be a valid user identifier, and $|T - T'|$ must fall in a valid interval; otherwise, stop the login request.

(2) Compute $A_i = h(ID_i, x)$ and $C1^* = h(Z_i / A_i \bmod p \oplus T)$.

(3) Compare $C1$ and $C1^*$. The login request is accepted only if they are the same; otherwise, the login request is rejected.

2.4 The password update phase

If the user wants to change his/her password PW_i to a new one PW_i', the password update steps are as follows:

(1) U_i sends ID_i and the new password PW_i' to the server in person or through a secure channel.

(2) The server computes $B_i' = g^{A_i h(PW_i')} \bmod p$.

(3) The server replaces B_i with B_i' in the smart card, and sends the card to the user. From now on, the user should use the new password PW_i' to login to the system.

3. The attack on the Lee-Chiu scheme

In this section, we shall show that the Lee-Chiu scheme is vulnerable to the impersonate attack. If the user U_i changes his/her password, an adversary can guess the password and thereby impersonate U_i to login to the system. An adversary guesses the password by the following steps:

(1) Eavesdrop the login message $m = \{ID_i, Z_i, C1, T\}$ when U_i uses the original password PW_i to login to the system, where

$$Z_i = A_i g^{A_i h(PW_i)} \bmod p \tag{1}$$

(2) Intercept another login message $m' = \{ID_i, Z_i', C1', T'\}$ when U_i uses the new password PW'_i to request another access. Here

$$Z_i' = A_i g^{A_i h(PW'_i)} \bmod p \tag{2}$$

(3) Compute $B = Z_i / Z'_i \bmod p$. Note that

$$B = Z_i / Z'_i = g^{A_i (h(PW_i) - h(PW'_i))} \bmod p \tag{3}$$

(4) The adversary guesses the original password and the new password as PW_{i_guess} and PW'_{i_guess}, respectively. Next, he/she computes D by

$$D = B^{(h(PW_{i\_guess}) - h(PW'_{i\_guess}))^{-1}} \bmod p \tag{4}$$

(5) Compute E by

$$E = Z_i / D^{h(PW_{i\_guess})} \bmod p \tag{5}$$

(6) Check whether $Z_i' \stackrel{?}{=} E \cdot D^{h(PW'_{i\_guess})} \bmod p$. If it is hold, the adversary guesses correctly. Otherwise, the adversary tries to guess another password by repeating the steps (4)-(6).

Discussion: Suppose that the adversary guesses the password correctly, then $PW_{i\_guess} = PW_i$ and $PW'_{i\_guess} = PW'_i$. By Eq.(3) and (4), the adversary obtains

$$\begin{aligned} D &= B^{(h(PW_{i\_guess}) - h(PW'_{i\_guess}))^{-1}} \bmod p \\ &= (Z_i / Z'_i)^{(h(PW_{i\_guess}) - h(PW'_{i\_guess}))^{-1}} \bmod p \\ &= g^{A_i (h(PW_i) - h(PW'_i))(h(PW_{i\_guess}) - h(PW'_{i\_guess}))^{-1}} \bmod p \\ &= g^{A_i} \end{aligned}$$

From Eq.(1), A_i will be obtained by $Z_i / D^{h(PW_i)} \bmod p$ since

$$\begin{aligned} & Z_i / D^{h(PW_i)} \bmod p \\ &= Z_i / g^{A_i h(PW_i)} \bmod p \\ &= A_i g^{A_i h(PW_i)} / g^{A_i h(PW_{i\_guess})} \bmod p \\ &= A_i \end{aligned}$$

Next, by using Eq.(2), the adversary can verify A_i by $Z_i' \stackrel{?}{=} A_i g^{A_i h(PW'_{i\_guess})} \bmod p$. If the equation holds, the exact passwords are obtained.

In general, to avoid having a password that keeps slipping off his/her mind, the user tends to choose one that is short and easily memorable, this leaves the adversary a very easy task on guessing the password. Though the adversary needs to guess both the original password and the new updated password, it is still an easy job on guessing because of the short bit-length of the passwords.

After guessing the password and obtaining A_i, the adversary can impersonate the legal user U_i and access the system by the following steps:

(1) Compute $Z_i = X \times A_i \bmod p$, where X is a random

number.

(2) Compute $C1 = h(X \oplus T)$, where T is the current time.

(3) Send the message $m = \{ID_i, Z_i, C1, T\}$ to the remote server.

On receiving the message, the server verifies the login request. The request will be accepted because

$$h((Z_i / A_i \bmod p) \oplus T) \\ = h(X \oplus T) \\ = C1$$

Therefore, if an adversary intercepts the login message before and after the user updates the password, he/she can guess the password and impersonate the legal user and login to the system.

4. The improvement scheme

The improvement scheme also has three phases: the registration phase, the login phase and the authentication phase. The registration phase and the password update procedures are the same as that of the Lee-Chiu scheme. The login phase and the authentication phase are as follows.

4.1 The login phase

U_i attaches the smart card to the card-reader and inputs identity ID_i and password PW_i*. The smart card does:

(1) Compute $B_i* = g^{A_i h(PW_i*)} \bmod p$.

(2) Compare B_i and B_i*. The procedure continued only if they are identical.

(3) Compute $Z_i = B_i \times h(A_i \oplus T) \bmod p$ and $C1 = h(B_i \oplus T)$, where T is the timestamp.

(4) Send the message $m = \{ID_i, Z_i, C1, T\}$ to the remote server.

4.2 The authentication phase

On receiving the login request m at time T', the server performs the following steps:

(1) Check ID_i and T. ID_i must be a valid user identifier, and $|T - T'|$ must be in a valid interval; otherwise, reject the login request.

(2) Compute $A_i = h(ID_i, x)$ and $C1* = h((Z_i / h(A_i \oplus T) \bmod p) \oplus T)$.

(3) Compare $C1$ and $C1*$. The login request is accepted only if they are the same; otherwise, the login request is rejected.

Security analysis

The security of the proposed scheme is based on the one-way hash function and the discrete logarithm problem. The proposed scheme need not maintain verification table; and has the merits such as resist replay attack, impersonate attack, and forgery attack.

The login message $m = \{ID_i, Z_i, C1, T\}$ contains the timestamp, a replay message can be detected by the server; thus the scheme can resists the replay attack.

Suppose that an adversary eavesdrops the message $m = \{ID_i, Z_i, C1, T\}$ and $m' = \{ID_i, Z_i', C1', T'\}$ when U_i login the system with the original password PW_i and the new password PW'_i, respectively. By

$$(Z_i / Z'_i)^{(h(PW_{i\_guess}) - h(PW'_{i\_guess}))^{-1}} \bmod p \\ = \frac{h(A_i \oplus T)}{h(A_i \oplus T')} \cdot g^{A_i (h(PW_i) - h(PW'_i))(h(PW_{i\_guess}) - h(PW'_{i\_guess}))^{-1}} \bmod p,$$

the adversary cannot obtains g^{A_i} unless he/she knows A_i and passwords. If an adversary tries to guess the password, it cannot work because the adversary cannot verify his/her guessing. Thus even if a user changes password, an adversary cannot guess the password and thereby impersonate U_i to access the system.

If an adversary wants to forge a message $m = \{ID_i, Z'_i, C1', T'\}$ to access the system, because of $Z'_i = B_i \times h(A_i \oplus T')$ and $C1' = h(B_i \oplus T')$, he/she cannot successfully login the system unless A_i is known. Thus the scheme can resist the forgery attack.

5. Conclusion

In 2005, Lee and Chiu proposed a remote authentication scheme that claimed to be secure. However, in this article, we shown that their scheme is vulnerable to the impersonate attack; an adversary can impersonate a legal user to login to the system. We also propose an improvement scheme to resist the impersonate attack.

***Acknowledgements*:** This work was supported partly by the National Science Council of the Republic of China under Contract Number NSC93-2218-E-150-013.

References

[1] Awasthi, A.K., and Lal. S.: 'A remote user authentication scheme using smart cards with Forward Secrecy', IEEE Transactions on Consumer Electronics,

2003, 49, (4), pp.1246-1248.

[2] Chang, C.C., and Wu, T.C.: 'Remote password authentication with smart card', IEE Proceedings. 1991, Part E 138, pp.165-168.

[3] Hwang, M.S., and Li, L.H.: 'A new remote user authentication scheme using smart cards', IEEE Transactions on Consumer Electronics, 2000, 46, (1), pp.28-30.

[4] Sun, H.M.: 'An efficient remote use authentication scheme using smart cards', IEEE Transactions on Consumer Electronics, 2000, 46, (4), pp.958-961.

[5] Lamport, L.: 'Password authentication with insecure communication', Communication of ACM, 1981, 24, pp.770-772.

[6] Wu, S.T., and Chieu, B.C.: 'A user friendly remote authentication scheme with smart cards', Computer and Security, 2003, 22, (6), pp.547-550.

[7] Lee N.Y., and Chiu, Y.C.: 'Improved remote authentication scheme with smart card', Computer Standards & Interfaces, 2005, 27, pp.177-180.

Word Sense Disambiguation and the Application on Internet Search

[1]Jason C. Hung, [2] Che-Yu Yang and [3]Timothy K. Shih
[1]Dept of Information Management,
Northern Taiwan Institute of Science and Technology, Taipei, Taiwan
[2,3]Department of Computer Science and Information Engineering,
Tamkang University, Taipei, Taiwan
[2]Email: yang@schummi.com

Abstract

All human languages have words that can mean different things in different contexts, such words with multiple meanings are potentially "*ambiguous*". The process of "deciding which of several meanings of a term is intended in a given context" is known as "*Word Sense Disambiguation (WSD)*". In this research, we investigated a hybrid method which combines "Wordnet ontology" and "concept co-occurrence statistics", to determine the degree of semantic relatedness between two lexically expressed concepts. Then we applied it in the WSD task, and the experimental results showed the good performance. Finally we give an example application of our work on Internet search. We demonstrated how to use Wordnet to expand search query, and how to utilize the WSD technique to filter out the noise in search results.

1. Introduction

Most of these textual materials retrieving approaches depend on a lexical match between words in users' requests and words in target objects. Typically only text objects that contain one or more common words with those in the users' query are returned as relevant. These word-based retrieval systems, however, are far from ideal ─ many objects relevant to a users' query are missed, and many unrelated objects are retrieved. Some researches show that fundamental characteristics of human verbal behavior result in these retrieval difficulties. Because of the tremendous variety in the words people use to describe the same meaning or concept (*synonymy*), people will often use different words from the author or indexer of the information, and relevant materials will therefore be missed. On the other hand, since a single word often has more than one meaning (*polysemy*), irrelevant materials will often be retrieved. Polysemy decreases retrieval precision by false matches and synonymy decreases the recall by missing true conceptual matches.

We try to overcome these problems by searching, or filtering textual information with its underlying concepts, rather than the keywords (word forms). All human languages have words that can mean different things in different contexts, such words with multiple meanings are potentially *ambiguous*. Such the process of deciding which of their several meanings is intended in a given context is known as *Word Sense Disambiguation (WSD)*. Accurate word sense disambiguation can lead to better results for information retrieval.

Because meaningful sentences are composed of meaningful words, any computer system that hopes to process natural languages as human do must have information about words and their meanings. One of the most widely known is the *Wordnet*, developed by George Miller [5] at Princeton University.

2. Wordnet

WordNet is a machine-readable dictionary (MRD) developed by George Miller and his colleagues at the Cognitive Science Laboratory at Princeton University. It is an online lexical database designed for use under program control, it provides a more effective combination of traditional lexicographic information and modern computing (George Miller, 1993 [5]). Synonymous words are grouped together into synonym sets, called *synsets*. Each such synset represents a single distinct sense or *concept*. For example, in Wordnet, the synset {car, auto, automobile, machine, motorcar} represents the concept of "4-wheeled motor vehicle; usually propelled by an internal combustion engine".

WordNet stores information about words that belong to four parts–of–speech: nouns, verbs, adjectives and adverbs. In Wordnet 2.0, there are 146,350 words organized in 111,223 synsets, approximate 20% of the words in WordNet are polysemous; approximate 40% have one or more synonyms, some 300 prepositions, pronouns, and determiners are given no semantic illustration in WordNet (George A. Miller, Richard Beckwith, 1993 [6]).

Wordnet database groups English nouns, verbs, adjectives, and adverbs into synsets that are in turn linked through semantic relations that determine word definitions and senses. WordNet 2.0 features a rich set of 333,612 relation links between synsets. Table 1 lists some semantic relations (links) in Wordnet (Fellbaum, C., 1998 [3]).

Table 1. Some semantic relations (links) defined in Wordnet.

Semantic Relation	Meaning	Example
Hypernym	*X* is a kind of *f(X)*	Apple *is a kind of* fruit
Hyponym	*f(X)* is a kind of *X*	Zebra *is a kind of* Horse
Holonym	*X* is a part/member of *f(X)*	Wheel *is a part of* a car
Meronym	*X* has part/member *f(X)*	Table *has part* leg
Antonym	*f(X)* is the opposite of *X*	Wet *is the opposite of* dry

The two most typical relations for nouns are *hyponymy* and *hypernymy*. For instance, {car, auto, automobile, machine, motorcar} are the hyponyms of {motor vehicle, automotive vehicle}, and {motor vehicle, automotive vehicle} are their hypernyms.

3. The Word Sense Disambiguation Method

The proposed WSD method is a hybrid approach that combines a knowledge-rich source, Wordnet, with a knowledge-poor source, the Internet (WWW) search. The intuition is that, generally, words that are conceptually related are more likely to be referred to together in a document (web page). Whereas words that are conceptually unrelated more seldom occurred together in a document. Thus, the *hit count* of the search results can be the evidence to judge the relatedness degree of each pair of concepts (synsets). The more hit count two concepts produce, the more relatedness they have.

The procedure to measure the relatedness between each candidate senses of the target word and each candidate senses of the compared word consists of two phrases. First, for the two words, we look it up at Wordnet to find all the candidate senses for each word, that is, we find every synsets which contain the word form of target word and the word form of compared form respectively. Then, we follow the hypernym links from the candidate synsets (senses) of each word forms, and get the "parent synsets" on the paths. Thus, we have all possible intended meanings and their synonyms for the two word forms now, as shown in figure 1.

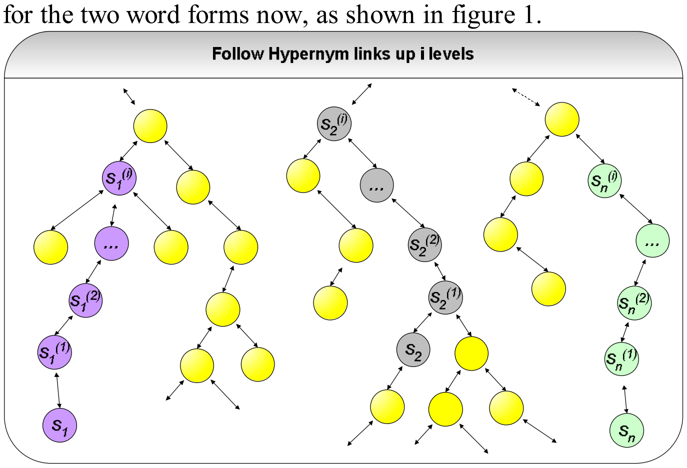


Figure 1. following hypernym links for each synsets

Second, we form every possible pair-combination of candidate senses (synsets), one sense chosen from a word, the other sense chosen from the other word. Then we use all the word forms in these two synsets and their "parent synsets" to form a query to search the Internet, and get a "hit count". This "hit count" would be normalized with the number of word forms in the search query. After processed with every pair of candidate senses, the pair of senses that has higher number of normalized hit count than other pairs are more likely to be the intended senses for target word in that context. We can utilize existing search engines which support complex Boolean search through its advanced search function, including brackets, AND, OR, NOT operators. The procedure to find the most appropriate sense for a target word by comparing all of its candidate senses with the candidate senses of a compared word is shown in figure 2.

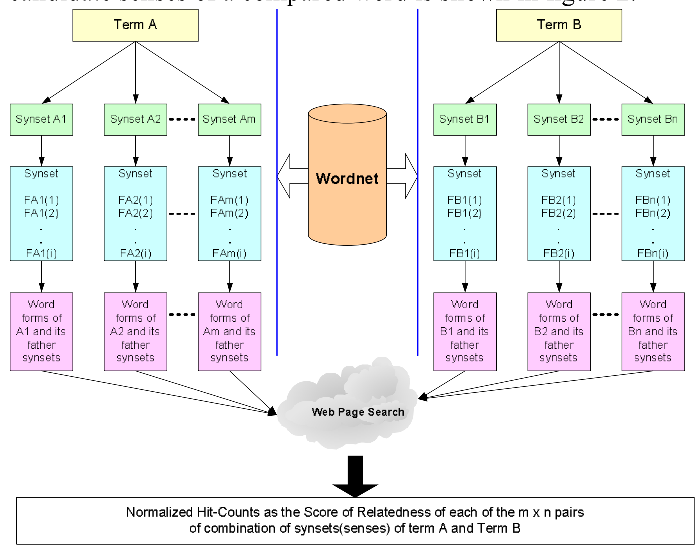


Figure 2. the procedure to find the most appropriate sense for a word..

The following is the algorithm to compute the relatedness between each pair of candidate senses of two words.

The Senses-Relatedness Calculation Algorithm:

Function:

Using "synsets (and the word forms in it)", "hypernym links" and "Boolean operators" to construct the representations of concepts (or senses) for given terms, and use the "hit counts" of Internet search as statistics to estimate the relatedness between each pair of senses.

Input:

1. two terms defined in Wordnet, one is the target term to disambiguate, the other is the compared term: *A* and *B*

2. level of the hypernym to trace up: *i*

Output:

Scores for each pair of candidate senses of the target term

Procedure:

1. For the input term *A* and *B*, find all the synsets that contain *A* and *B* respectively. If there are *m* senses defined for term *A* in Wordnet and *n* senses defined for term *B*, then we get set of synsets $S_A = \{ A_1, A_2, A_3 \ldots, A_m \}$ and $S_B = \{ B_1, B_2, B_3 \ldots, B_n \}$, where A_m *and* B_n are synsets;

2. For each A_m in S_A and B_n in S_B, we follow its hypernym semantic link up to *i* levels to find its hypernymy synsets on the path, then we get $H_{Am} = \{ A_m^{(1)}, A_m^{(2)} A_m^{(3)} \ldots, A_m^{(i)} \}$, where $A_m^{(1)}$ is the parent synset of A_m, $A_m^{(2)}$ is the parent synset of $A_m^{(1)}$, and so on. Also we get $H_{Bn} = \{ B_n^{(1)}, B_n^{(2)} B_n^{(3)} \ldots, B_n^{(i)} \}$, where $B_n^{(1)}$ is the pBrent synset of B_n, $B_n^{(2)}$ is the parent synset of $B_n^{(1)}$, and so on,

Thus, for all senses of term A, $S_A = \begin{matrix} A_1 \\ A_2 \\ A_3 \\ \vdots \\ A_m \end{matrix}$, we get

$$H_A = \begin{matrix} H_{A1} \\ H_{A2} \\ H_{A3} \\ \vdots \\ H_{Am} \end{matrix} = \begin{matrix} A_1^{(1)}, A_1^{(2)}, A_1^{(3)} & A_1^{(i)} \\ A_2^{(1)}, A_2^{(2)}, A_2^{(3)} & A_2^{(i)} \\ A_3^{(1)}, A_3^{(2)}, A_3^{(3)} & A_3^{(i)} \\ \vdots & \\ A_m^{(1)}, A_m^{(2)}, A_m^{(3)} & A_m^{(i)} \end{matrix}$$

; for all senses of term B,

$S_B = \begin{matrix} B_1 \\ B_2 \\ B_3 \\ \vdots \\ B_n \end{matrix}$, we get $H_B = \begin{matrix} H_{B1} \\ H_{B2} \\ H_{B3} \\ \vdots \\ H_{Bn} \end{matrix} = \begin{matrix} B_1^{(1)}, B_1^{(2)}, B_1^{(3)} & B_1^{(i)} \\ B_2^{(1)}, B_2^{(2)}, B_2^{(3)} & B_2^{(i)} \\ B_3^{(1)}, B_3^{(2)}, B_3^{(3)} & B_3^{(i)} \\ \vdots & \\ B_n^{(1)}, B_n^{(2)}, B_n^{(3)} & B_n^{(i)} \end{matrix}$;

3. Form combinations of every possible pairs of synsets selected from A and B respectively, then we get (m n) pairs of senses pairs, as follows:

$$Q_{A,B} = \begin{matrix} (A_1 \; H_{A_1}) & (B_1 \; H_{B_1}) \\ (A_1 \; H_{A_1}) & (B_2 \; H_{B_2}) \\ (A_1 \; H_{A_1}) & (B_3 \; H_{B_3}) \\ \vdots \; \vdots & \vdots \; \vdots \\ (A_1 \; H_{A_1}) & (B_n \; H_{B_n}) \end{matrix}, \begin{matrix} (A_2 \; H_{A_2}) & (B_1 \; H_{B_1}) \\ (A_2 \; H_{A_2}) & (B_2 \; H_{B_2}) \\ (A_2 \; H_{A_2}) & (B_3 \; H_{B_3}) \\ \vdots \; \vdots & \vdots \; \vdots \\ (A_2 \; H_{A_2}) & (B_n \; H_{B_n}) \end{matrix} \quad \begin{matrix} (A_m \; H_{A_m}) & (B_1 \; H_{B_1}) \\ (A_m \; H_{A_m}) & (B_2 \; H_{B_2}) \\ (A_m \; H_{A_m}) & (B_3 \; H_{B_3}) \\ \vdots \; \vdots & \vdots \; \vdots \\ (A_m \; H_{A_m}) & (B_n \; H_{B_n}) \end{matrix}$$

4. Using word forms in the synsets in $Q_{A,B}$ to form (m n) search queries to search the Internet, and use the hit counts of match web pages divided by the number of word forms in each query as the *normalized score* for each query.

To formalize the WSD procedure, we have developed an approach. In our approach, we starts by selecting a *windows of context* that consists of target words and some number of context words to the left and right that are known to Wordnet., and a set of candidate senses is identified for each content word in the window based on the sense inventory in the Wordnet. Assume that the window of context consists of 2n+1 words denoted by w_i, $-n \leqq i \leqq +n$, where the target word is w_0. Further let $| w_i |$ denote the number of candidate senses of word w_i, and let these senses be denoted by $s_{i,j}$, $1 \leqq j \leqq | w_i |$.

After the candidate senses are determined, we measure the relatedness of the candidate senses of the target word to those of the surrounding words in the window of context. From this a score is computed for each sense of the target word, these scores specified how related it is to the senses of the words in the window of context.

We assign to each possible sense k of the target word a $SenseScore_k$ computed by adding together the relatedness scores obtained by comparing the senses of the target word in question with every sense of every non–target word in the window of context. The *SenseScore* for sense $s_{0,k}$ (which is the k[th] sense of the target word) is computed as follows:

$$SenseScore_k = \sum_{i=-n}^{n} \sum_{j=1}^{|w_i|} relatedness(s_{0,k}, s_{i,j}), i \neq 0 \quad \text{……. (1)}$$

The sense with the highest *SenseScore* is judged to be the most appropriate sense for the target word.

The reason we include the hypernym synsets of the target synset and the compared synset is that: we found that, in Wordnet, sometimes there is only one single word form without other synonym word form(s) in a synset. In the below example, only the first sense of the word form "eagle" has two word forms in it ("eagle" and "bird of Jove"). If we use only target and compared synsets themselves to construct the search query (without their hypernym synsets), then the search results (hit counts) formed by senses 2, 3, 4 in this example would be identical. This should be prevented from happening.

Sense 1
eagle, bird of Jove -- (any of various large keen-sighted diurnal birds of prey noted for their broad wings and strong soaring flight)
@-> bird of prey, raptor, raptorial bird
@-> bird
Sense 2
eagle -- ((golf) a score of two strokes under par on a hole)
@-> score
@-> numbe
sense 3
eagle -- (a former gold coin in the United States worth 10 dollars)
@-> coin -- (a metal piece (usually a disc) used as money)
@-> coinage, mintage, specie, metal money -- (coins collectively)
sense 4
eagle -- (an emblem representing power; "the Roman eagle")
@-> emblem, allegory -- (a visible symbol representing an abstract idea)
@-> symbol, symbolization, symbolisation, symbolic representation

From not only the word form(s) in the synsets that contain the original word form, but also the hypernym synsets of the original synsets (the four senses), we now have more clues/alternatives to represent the different senses/concepts of a word form now.

4. Evaluation

Most of other works focus on a selected set of a few words. Generally, these selected words bear small number of senses of very different meaning, thus for which their algorithm could easily gather enough evidence to disambiguate the words. On the contrary, our experiment was tested on ALL the nouns in a subset of an unrestricted public domain corpus, thus has to make fine-grained distinctions among all the senses in WordNet. This makes the comparison with other works somewhat difficult.

4.1 Experiment setup

The SemCor corpus, created by the Princeton University, is a subset of the English Brown corpus containing about 700,000 running words. In SemCor all the words are tagged by PoS, and more than 200,000 content words are also lemmatized and sense-tagged according to Princeton WordNet. More in detail, the SemCor corpus is composed of about 352 texts. In some texts all the open class words (nouns, verbs, adjectives, and adverbs) are annotated with PoS, lemma and sense, such example as figure 3. The "all-words" component of SemCor has about 359,732 tokens among which almost 192,639 are semantically annotated.

Nothing in English has been ridiculed as much as the ambiguous use of words.

```
------------------------------------------------------------------------------------------------
<contextfile concordance=brown>
<context filename=br-r05 paras=yes>
<p pnum=1>
<s snum=1>
<wf cmd=done pos=NN lemma=nothing wnsn=1
lexsn=1:23:00::>Nothing</wf>
<wf cmd=ignore pos=IN>in</wf>
<wf cmd=done pos=NN lemma=english wnsn=1
```

```
lexsn=1:10:00::>English</wf>
<wf cmd=done pos=VBZ ot=notag>has</wf>
<wf cmd=done pos=VBN ot=notag>been</wf>
<wf cmd=done pos=VB lemma=ridicule wnsn=1
lexsn=2:32:00::>ridiculed</wf>
<wf cmd=done pos=RB ot=complexprep>as_much_as</wf>
<wf cmd=ignore pos=DT>the</wf>
<wf cmd=done pos=JJ lemma=ambiguous wnsn=1
lexsn=3:00:04::>ambiguous</wf>
<wf cmd=done pos=NN lemma=use wnsn=1 lexsn=1:04:00::>use</wf>
<wf cmd=ignore pos=IN>of</wf>
<wf cmd=done pos=NN lemma=word wnsn=1 lexsn=1:10:00::>words</wf>
<punc>.</punc>
```

Figure 3. Example sentences and their tagged format in Semcor.

Semcor inventories various genders of text. We selected four texts from SemCor: br-a01, br-b20, br-j09 and br-r05.

4.2 Experimental Results

To remind one important fact again - our experiment was tested on ALL the nouns in a subset of an unrestricted public domain corpus (semcor), thus has to make fine-grained distinctions among all the senses in WordNet.

The "baseline" precision is the possibility of hit the correct target sense by wild guess. For example, the word form "eagle" has four different senses, so the baseline precision of this word is 1/4 (25%). Thus the baseline precision values in the figures are calculated by:

$$\text{Baseline}=\frac{1}{\text{average number of senses of all the word forms to be disambiguated}}$$

The figure 4 lists the precision of WSD against different numbers of hypernym synsets inclusion in the query string.

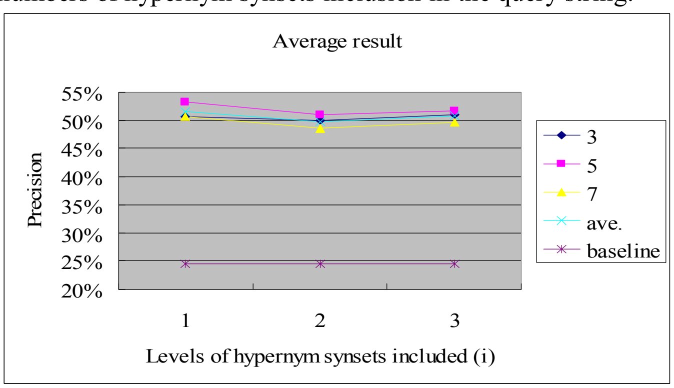


Figure 4. the average precision against different numbers of hypernym synsets inclusions

Firgure 5 lists the precision of WSD against different size of window of context.

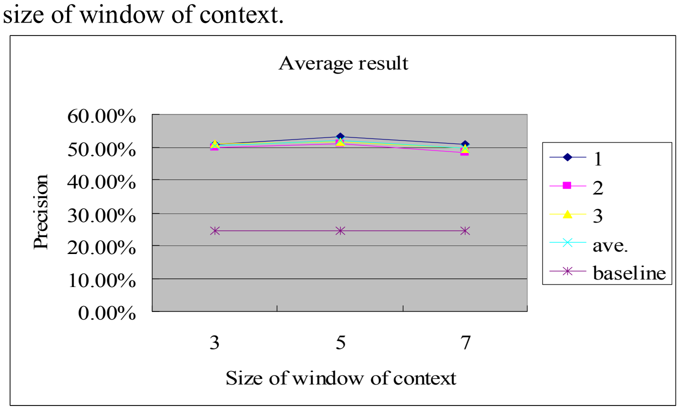


Figure 5. the average precision against different size of window of context

4.3 Discussion

It is of some interest that, in principle, of course, there is no limit to the number of levels an inheritance system can have. Lexical inheritance systems, however, seldom go more than ten levels deep, and the deepest examples usually contain technical levels that are not part of the everyday vocabulary. For example, in the hierarchy –

pony -- (any of various breeds of small gentle horses usually less than five feet high at the shoulder)
@-> horse, Equus caballus
@-> equine, equid
@-> odd-toed ungulate, perissodactyl, perissodactyl mammal
@-> ungulate, hoofed mammal
@-> placental, placental mammal, eutherian, eutherian mammal
@-> mammal
@-> vertebrate, craniate
@-> chordate
@-> animal, animate being, beast, brute, creature, fauna
@-> organism, being
@-> living thing, animate thing
@-> object, physical object
@-> entity

There are total fourteen levels, and about half of them are technical lexemes, not everyday vocabulary. In fact, the number of hierarchy levels is limited in Wordnet already, because advocates of redundant storage of the information associated with these concepts point out that the more generic information would be repeated over and over in a redundant system, so each additional level would put an increasingly severe burden on lexical memory. But there are still some words that more often being referred to, while some others seldom, in the different levels of hierarchy.

We think because of this property of Wordnet hierarchy, the different number of hypernym levels included (the "*i*" value) when searching the Internet for concept co-occurrence dose NOT affect the result in a consistent manner. We can observe this in the evaluation diagrams.

In this example, look at the three adjacent hypernym/hyponym synsets: {"*pony*"}, {"*horse", "Equus caballus"*} and {"*horse", "Equus caballus"*}. For the synset {"*pony*"}, it may benefit from including its hypernym synset {"*horse", "Equus caballus"*}; whereas for the synset {"*horse", "Equus caballus"*}, it may NOT benefit from including its hypernym synset {"*equine", "equid*"}. In the same way, it does NOT make lots of difference for the synset {"*pony*"} to include one level, or two levels or three levels of hypernym synsets.

About the size of window of context, what is the optimum context-size for disambiguating using our WSD method is an important issue. People could assume that the more context word being compared, the better the disambiguation results would be. Nevertheless the actual nature of each text is for sure an important factor that is difficult to measure. How far the useful context words usually appear around the target to-be-disambiguated word is unsteady, depending on the properties of the different text source. Thus the best window size depends on the natural property of the text source.

5. Facilitate the Internet Search

In this section we proposed a "prototype" which uses Wordnet to facilitate query expansion, and uses WSD technique to filter out the noise in search results.

There are many publicly available search engines, but users are not necessarily satisfied with the different formats for inputting queries and quality of retrieved information. They are - *"How to select most proper words"* and "*the inability to find relevant information.*"

5.1 Query Expansion According to Word Sense

One difficult problem in text IR is how to deal with multiple terms that refer to the same concept (synonym). If a query interface does not take this into account when processing search terms, then its search results will be incomplete.

A query expansion-enabled interface will take as input a given search term, look for synonyms in the controlled vocabulary, and return documents that match either the search term or any of its synonyms.

We carry out the query expansion by the following method. For the words in user query, the system looks up in Wordnet for the senses of these word forms, and returns the part-of-speech, glosses and synonyms of each sense. User can browse the synonymy information according to for types of part-of-speech, and assign the intended meanings for each term.

To deal with the difficulty in choosing search terms, for example, if a user submits a query consists of "book", "train" and "ticket". After assigning a sense of interest to each search term, the system can expand the query to include the synonyms terms of that sense. In this example, an expanded query like the following one will be formed.

Original query: *book AND train AND ticket*
Expanded query: *（reserve OR hold OR book） AND (train OR "rail train") AND (ticket)*

Hence the search will return the documents that match not limited to the original search terms inputted by user, but any of its synonyms. By this semi-automatic query expansion technique, we solve the problem of "synonym", thus can improve the overall "recall" of the search.

5.2 Noise Filter-Out Using WSD

The polysemy problem leads to the false match. To solve the polysemy problem in information retrieval requires the disambiguation of word meanings when separate concepts are expressed by the same term.

We apply our word sense disambiguation method to solve the polysemy problem here. The idea is that, after search with semantically expanded query, there may still be irrelevant web pages (noise) mixed up in the search results; thus, we carry out WSD process to the words (in returned pages) that match the words in user's query. After disambiguating these matched words in the web pages, we compare their senses with the ones that the user had assigned during the query expansion phrase. Then we can eliminate pages that are not conceptually match from search results, solve the polysemy problem, thus improve the precision of the search.

The whole system flow of our proposed mechanism to improve search engine is as figure 6.

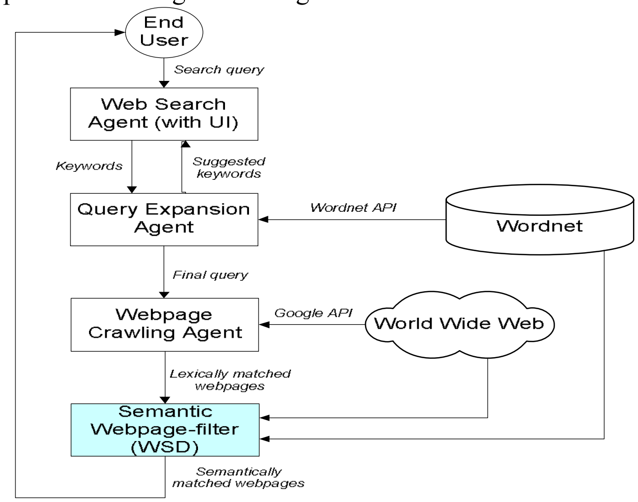


Figure 6. System architecture of improved "semantic" search engine.

6. Related Work

In some Wordnet based WSD approaches, semantic distance is calculated using the edge counting principle. The measurement usually used (Rada et al. 1989 [11], and Lee et al. 1993) is the shortest path between concepts.

(Hirst and St-Onge's, 1998 [7]) measure semantic relatedness based on that, two lexicalized concepts are semantically close if their WordNet synsets are connected by a path that is not too long and that "does not change direction too often".

The use of density is based on the observation that words in a more densely part of the hierarchy are more closely related than words in sparser areas (Agirre and Rigau, 1996 [1]). They proposed the *conceptual density* concept for WSD.

(Rosso, P., Masulli, F. and Buscaldi, D., 2003 [13]) proposed a method for the resolution of lexical ambiguity relying on the use of the wide-coverage noun taxonomy of WordNet and the notion of conceptual distance among concepts, captured by a conceptual density formula developed for this purpose.

(Leacock and Chodorow, 1998 [9]) also rely on the length of the shortest path between two synsets for their measure of similarity. However, they limit their attention to IS-A-KIND-OF links and *scale* the path length by the overall depth *D* of the taxonomy.

There some hybrid approaches, that combine thesaurus, with corpus statistics (Resnik, 1995 [12]; Lin, 1998; Jiang and Conrath, 1997 [8]). Resnik defines the similarity of two concepts defined in WordNet to be the maximum *information content* of their lowest super-ordinate. The information content of a concept relies on the probability of encountering an instance of the concept. (Jiang and Conrath, 1997 [8]) uses the notion of information content in the form of the conditional probability of encountering an instance of a child-synset when given an instance of a parent-synset.

(Mihalcea and Moldovan, 1999 [10]) first determine the most common sense-pairs. Subsequently, verb-noun pairs are disambiguated by taking the first t possible senses of the words and calculating "conceptual density" of the pairs by examining the WordNet glosses of the sub-hierarchies. This then ranks each pair of senses by looking at the noun-context of the verb and comparing it with the given noun. The algorithm appeared to have been thoroughly tested only for pairs of words.

(Chua, S. and Kulathuramaiyer, N., 2004 [2]) employs noun synonyms and word senses for feature selection to select terms that are semantically representative of a category of documents. The categorical sense disambiguation extends the use of WordNet.

Information retrieval using word senses is emerging as a good research challenge on semantic information retrieval. (Kim, S. B., Seo, H. C. and Rim, H. C., 2004 [4]) proposed a method using word senses in information retrieval: root sense tagging method.

7. Conclusion

The proposed word sense disambiguation method is a hybrid approach that combines a knowledge-rich source, Wordnet, with a knowledge-poor source, the Internet (World Wide Web) search. The words in context are paired, and each word is disambiguated by searching the Internet with queries formed by using different senses and their hypernymy senses of each of the two words. The senses are then ranked according to the number of search hits. The intuition is that, generally, words that are conceptually related are more likely to be referred to together in a document (web page). Whereas words that are conceptually unrelated more seldom occurred together in a document.

In out experiment, we found that different number of hypernym levels included (the "*i*" value) when searching the Internet for concept co-occurrence dose NOT affect the result in a consistent manner. Because of the property that there are some words that more often used or referred to by people, while some others seldom, in the different levels of hierarchy. On the other hand, the size of "window of context" when disambiguating the target word seems to be better at size of five (two words left and two words right) for our WSD method. But we believe it still mainly depends on how far the useful context words usually appear around the target word.

Most of existing search engines depend on a lexical match between words in users' requests and words in database objects. By using a semi-automatic way that employs Wordnet to prompt/suggest senses for word forms in the search query to user, we demonstrated how to reduce the effect of "synonymy" on the "recall" of the search.

On the other hand, the problem of morphologically identical terms may refer to separate concepts (polysemy), leads to the false match. Our solution is that, after search, there may still be irrelevant web pages (noise) mixed up in the search results; thus, we carry out WSD process to the words (in returned pages) that match the words in user's query. After disambiguating these matched words in the web pages, we compare these senses with the ones that the user had assigned during the query expansion phrase. Then we can eliminate pages that are not conceptually match from search results, solve the "polysemy" problem, thus improve the "precision" of the search.

Reference

[1] Agirre and Rigau. "Word sense disambiguation using conceptual density." The 16th International Conference on Computational Linguistics, , Copenhagen, 1996.

[2] Chua, S. Kulathuramaiyer, N. "Semantic Feature Selection Using WordNet." In Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence (WI 2004), page(s): 166 – 172, 2004.

[3] Fellbaum C. An Electronic Lexical Database. MIT Press, Cambridge, Mass., 1998..

[4] Kim, S. B., Seo, H. C. and Rim, H. C. "Information retrieval using word senses: root sense tagging approach." In Proceedings of the 27th annual international conference on Research and development in information retrieval (SIGIR '04), Sheffield, United Kingdom, July 25 - 29, 2004.

[5] Miller, George A. "WordNet: A Lexical Database." Comm. ACM, Vol. 38, No. 11, 1993, pp. 39-41.

[6] Miller, George and Beckwith, Richard. "Introduction to WordNet: An On-line Lexical Database." Revised 1993.

[7] Hirst and D. St-Onge. "Lexical chains as representations of context for the detection and correction of malapropisms." In WordNet: An electronic lexical database. MIT Press, 1998.

[8] Jiang, J.J. and Conrath D.W. "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy." ROCLING X International Conference on Research in Computational Linguistics, Taiwan, 1997.

[9] Leacock and Chodorow. "Combining local context and WordNet similarity for wordsense identification." In Fellbaum 1998, 265-283.

[10] Moldovan, D., and Mihalcea, R. "Using WordNet and lexical operators to improve internet searches." IEEE Internet Computing, vol.4 no.1, pp. 34-43, 2000.

[11] Rada, R., H. Mili, E. Bicknell, and M. Bletner. "Development and Application of a Metric on Semantic Nets." IEEE Transactions on Systems, Man and Cybernetics, Vol. 19, No. 1, 17-30, 1989.

[12] Resnik, P. "Using Information Content to Evaluate Semantic Similarity in a Taxonomy." Proceedings of the 14th International Joint Conference on Artificial Intelligence, Vol. 1, 448-453, Montreal, August 1995.

[13] Rosso, P., Masulli, F. and Buscaldi, D. "Word sense disambiguation combining conceptual distance, frequency and gloss." In Proceedings of International Conference on Natural Language Processing and Knowledge Engineering, page(s): 120-125, Oct. 2003

Agent-based Transactions for Home Energy Services

Ronggong Song, Larry Korba, and George Yee

Institute for Information Technology
National Research Council of Canada
Ottawa, Ontario K1A 0R6, Canada
{Ronggong.Song, Larry.Korba, George.Yee}@nrc.ca

Abstract—**Current home energy management systems (HEMS), which use the existing home networks and Internet as the communication infrastructure, are both inexpensive and offer definite consumer advantages. However, most of them don't support e-services such as reporting and dealing with monthly expenses for services like power, gas, and water. A key deterrent for the handling of the e-services in this environment is that the data may be subject to a variety of active and passive attacks if the system is not designed properly. This paper describes a lightweight security architecture for e-services related to home energy services. The system is intended to protect against both inside and outside attacks by using the existing HEMS and combining the tamper-resistant devices with Internet security techniques.**

1 Introduction

Home energy management systems (HEMS) [1, 2, 3] are designed to control home appliances automatically for the purpose of both convenience and saving energy. However, most of them do not support e-services such as reporting the monthly expenses for services and dealing with the billing statements of customers. Not having automatic e-reporting built into HEMS is also unfortunate due to the inefficiency and inconvenience of having to use a separate billing procedure (often involving manual processing). Furthermore, security technologies such as SSL [4] or VPN used in the current HEMS such as [3] are insufficient for providing secure communications required for e-reporting between energy devices and the service providers. This is because VPN and SSL services are both controlled by the customers leading to the possibility that customers may easily make a number of possible active attacks on the e-reporting data (e.g. changing the amount of the energy consumption).

An additional challenge is that providing a secure architecture to handle e-services for home energy services must deal with the fact that the devices embedded within the meter normally have limitations computationally in order to keep costs down.

In order to solve the above problems, we propose a lightweight security architecture for the e-reporting and billing of home energy services based on tamper-resistant technology and SSL (or TLS [5]) technology. The security architecture consists of three parts: a meter reading device (MRD), a middleware agent, and a service provider's server. The middleware agent is located on the customer's computer or home network gateway. It registers the customer to the service provider's server and forwards the e-reporting protocol messages between the MRD and the server. The e-reporting protocol could be initiated by the middleware agent or the service provider's server depending on different implementation. Communications between the MRD, the agent, and the server are protected with a combination of security mechanisms (See Section 3.3). The meter reading device is a tamper-resistant device. It uses a keyed hash function with a secret key shared between the MRD and server. The data transferred from the MRD to the provider must be accompanied with a hash result for the data integrity and authentication protection against some inside and outside active attacks (e.g. replay and modification). The data is then forwarded to the provider by the middleware agent through an SSL secure channel in order to provide data confidentiality, authentication, and integrity protection against outside attacks. In addition, since the security architecture is based on the existing home network, Internet, and SSL security technology, the new system only needs new hardware in the form of the MRD, new software – the middleware agent, and some new security protocols.

The rest of the paper is organized as follows. A home energy management system [3] is briefly reviewed, and the security requirements for the e-reporting are analyzed in the next section. In Section 3, the security architecture is proposed for the e-reporting. In Section 4, the new security protocols are described for the security architecture. In Section 5, some characteristics of the new secure architecture are summarized. In Section 6, the security of the new protocols is analyzed. Finally, concluding remarks are given in Section 7.

2 Review of Home Energy Management System

2.1 Home Energy Management System

Based on [3], a home energy management system comprises an in-house system connected to service control servers via the Internet. The in-house system includes a home network and network adapters for the energy service appliances. The service control servers include the service provider's server, the information provider's server, and the energy control server. The home network and Internet provide the network communication infrastructure for the communications between the energy appliances and service control servers. Figure 1 depicts a home energy management system. In this paper, we only talk about the security architecture for the e-reporting and billing among the energy appliances, the customers, and the service providers. The service providers include a gas service provider, a water service provider, and a power service provider.

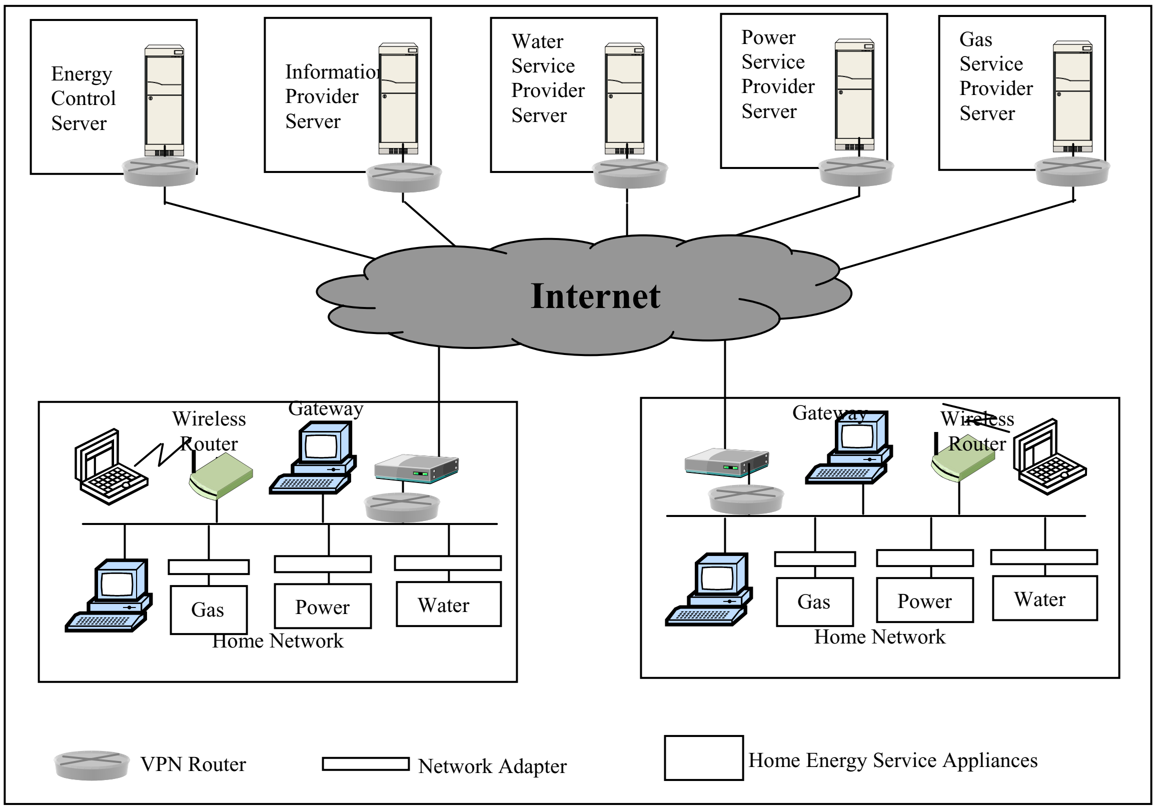


Fig. 1. A home energy management system

2.2 Security Analysis of HEMS

The current home energy management systems use VPN or SSL as their security architecture. VPN provides the security communication protection between the home network domain and the server domain (for instance, the above energy control server domain). SSL provides the security communication protection between the customer's application and the remote server. However, these two security technologies are not enough to protect the security communications between the energy appliance and the service provider since the customers usually have full control over their home network domain, i.e., the customers can easily make some inside attacks on the communications if the communications are not protected properly including some attacks made by outside hackers. In addition, the conflictive benefits between the customers and service providers may encourage the customers to make these attacks, for instance, changing the amount of energy consumption for the e-reporting. Thus we need other security technologies to protect the communications.

In addition, since SSL has provided the security communications between the application and the service provider server against the outside attacks, a confidentiality protection is not very important on the home network for the reporting data transferred from MRD to the provider's server but the integrity and authentication protection may become very important against some inside and outside active attacks like replay and modification. This special situation gives an opportunity to make a lightweight security architecture for it.

3 Security Architecture

The security architecture for the e-reporting of HEMS consists of three parts: a meter reading device, a middleware agent, and a service provider's server.

3.1 Meter Reading Device

The meter reading device is a tamper-resistant device which is embedded with the meter together. The device directly connects with a network adapter to communicate with outsider through the home network. Figure 2 depicts the functionality of MRD.

In MRD, *h* is a keyed hash function and *P* is a control processing part for verifying whether or not the input data is correct. The secret key is stored in the hiding area of MRD. It is a shared key with the service provider, i.e. except the provider, other people including the customer knows nothing about the key. The output data from MRD to the provider is hashed by the hash function with the secret key for the data integrity and authentication protection. The provider verifies the data received from the MRD using the hash function and the shared secret key. Likewise, the data transferred from the provider to the MRD should be verified in the control processing part. In order to make the system stronger each MRD should choose a different secret key.

The functionality of the tamper-resistant device embedded in MRD is similar to a simple smart card. We could use a simple smart card technology for it. The expensive strong tamper-resistant devices are not necessary for this system since the main purpose of the tamper-resistant device is to protect the shared secret key. In addition, the providers could physically check the randomly selective meters and devices, especially if they find some suspicious cases.

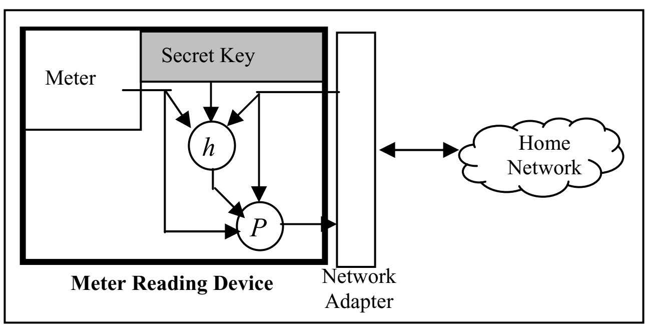


Fig. 2. The Functionality of MRD

3.2 Middleware Agent

The middleware agent could be located on the customer's computer or home network gateway. After installing the middleware agent, the customer then uses it to register his service account in the service provider's server and set up a schedule for reporting the mount of energy consumption and billing the statement for payment. The e-reporting and billing protocol could be initiated with the middleware agent or the service provider's server. For the former, the customer should set up the time and period to wake up the protocol implemented in the middleware agent. For the latter, the customer should set up the initiating (waking up the protocol) condition as getting the initiating message from the service provider's server. The middleware agent forwards the messages between MRD and the server during the e-reporting and billing. Figure 3 depicts the state of the middleware agent.

The communications among MRD, the middleware agent, and the service provider's server are protected by a combination of secure mechanisms. Figure 4 depicts the security channels where the SSL (or TLS) secure channel protects all communications between the agent and server in the transport layer against outside attacks for the data confidentiality and integrity. The message authentication code (MAC) channel protects the communications between MRD and the service provider's server in the application layer against the inside and outside active attacks. The SSL secure channel will wrap the integrity secure channel between the agent and server. The SSL secure channel is an option depending on the requirements.

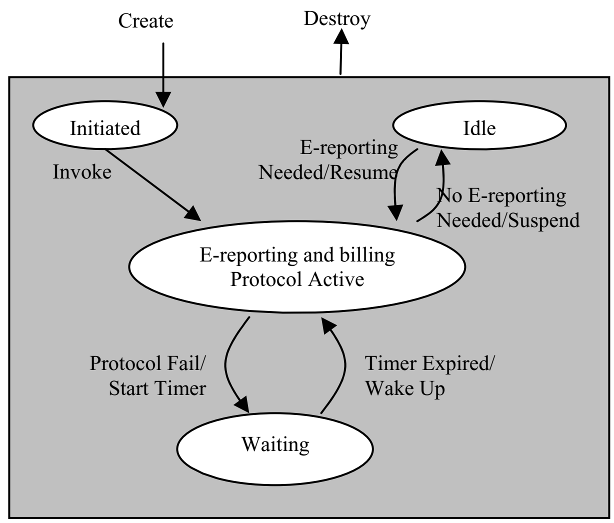


Fig. 3. The State of Middleware Agent

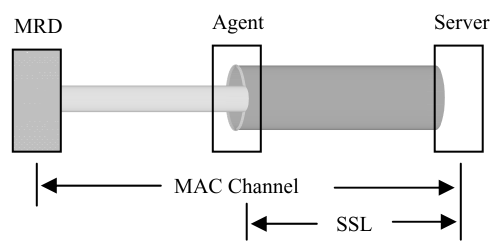


Fig. 4. The Security Channels

3.3 Service Provider's Server

The service provider's server is a normal application server but it is implemented with the SSL secure technology in the transport layer and the MAC authentication functionality in the application layer. The integrity and authentication processing will be described in the following security protocols.

4 Security Protocols

In order to implement the security functions, the security architecture has two protocols: registration protocol and e-reporting and billing protocol.

4.1 Terminology and Notations

Terminology and notations used in the protocols are defined as follows.

- *C*: a customer
- *A*: a middleware agent
- *P*: an energy service provider
- *MRD*: a meter reading device
- ID_C: customer *C*'s identity
- ID_{MRD}: an identity number of MRD
- $Account_C$: customer *C*'s account number
- N_P: a nonce made by energy service provider *P*
- $Time_P$: a time stamp made by provider *P*
- *SK*: a secret key embedded in MRD
- *H*(): one-way hash function

4.2 Registration Protocol

All customers are required to register with the energy service provider's server and set up their service accounts in the server when they first install the middleware agent in their computer. Figure 5 depicts the message flow of the registration protocol. In the protocol, the communications between the middleware agent and the server is protected with the SSL secure channel in the transport layer.

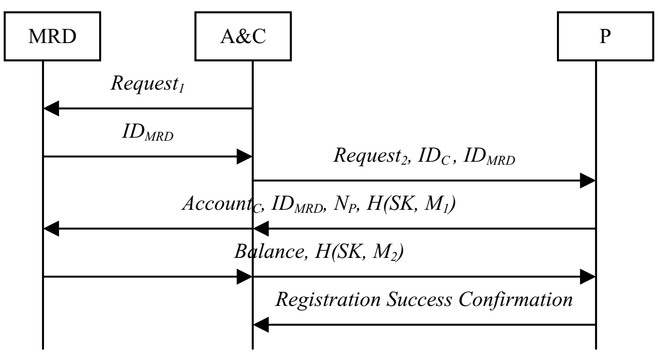


Fig. 5. The Registration Protocol

The registration protocol is described as follows.

Step 1: The agent *A* sends a request message ($Request_1$) to MRD for its identity number.

Step 2: MRD responds to the request message giving its identity number (ID_{MRD}).

Step 3: The agent *A* then sends another request message ($Request_2$) to the energy service provider's server with the customer's personal information (ID_C, e.g., Name) and MRD's identity number (ID_{MRD}) together. In this step, all messages are protected with the SSL secure channel.

Step 4: After receiving the above messages, the server first verifies if the customer's personal data and MRD's identity are correct through its database or social network. If they are correct, the server then sets up an account for the customer and sends the message [$Account_C$, ID_{MRD}, N_P, $H(SK, M_1)$] to the agent *A* where *SK* is a shared secret key between the service provider and MRD, and M_1 = [$Account_C$, ID_{MRD}, N_P].

The agent *A* then verifies and forwards the message to MRD.

Step 5: MRD verifies if the message [ID_{MRD}, $H(SK, M_1)$] is correct. If it is correct, MRD then sends the energy consumption balance (*Balance*) and keyed hashing result [$H(SK, M_2)$] to the agent *A* where M_2 = [$Account_C$, ID_{MRD}, N_P, *Balance*].

The customer then checks if the balance is correct. If it is correct, the customer could let the agent *A* forwards the message to the server.

Step 6: The server then verifies if the message is correct. If it is correct, the server then sends a registration success confirmation message to the agent *A*.

4.3 E-reporting and billing Protocol

After registration, the customer and provider could set up the time and period for initiating the e-reporting and billing protocol. The e-reporting and billing protocol could be initiated with either the middleware agent or the service provider's server depending on the different implementation. We design two different e-reporting and billing protocols for them. Their features are shown in Section 5.

The first e-reporting and billing protocol is initiated with the middleware agent. For this implementation, the middleware agent is like client-side software which makes the implementation more simple and low cost. Figure 6 depicts the message flow of the e-reporting and billing protocol.

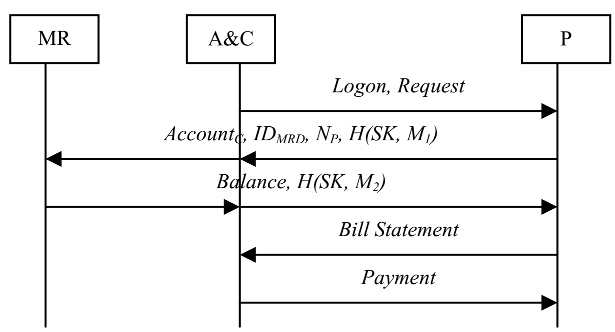


Fig. 6. The E-reporting and billing Protocol

The protocol is described as follows.

Step 1: The agent *A* first logs on the server and requests the bill statement.

Step 2: After verifying the user's name and password, the server sends the message [$Account_C$, ID_{MRD}, N_P, $H(SK, M_1)$] to the agent A where M_1 = [$Account_C$, ID_{MRD}, N_P].

After receiving the messages, the agent A first verifies if the message [$Account_C$, ID_{MRD}, N_P] is correct. If it is correct, the agent A then forwards the message to MRD.

Step 3: MRD verifies if the message [ID_{MRD}, $H(SK, M_1)$] are correct. If it is correct, MRD then sends the current energy consumption balance (*Balance*) and hashing result [$H(SK, M_2)$] to the agent A where M_2 = [$Account_C$, ID_{MRD}, N_P, *Balance*].

The customer then checks if the balance is correct. If it is correct, the customer could let the agent A forwards the message to the server.

Step 4: The server first verifies if the message is correct and then calculates the balance of the energy consumption in the last period. Finally, the server makes a bill statement and sends it to the agent A and customer C.

Step 5: The customer C could directly pay the bill through the agent A using his/her credit card, or make the payment through other Online Banking.

The second e-reporting and billing protocol is initiated with the service provider's server. For this implementation, the middleware agent works like server-side software or client-side software with push technologies which requires the customer to install it to an application server. Figure 7 depicts the message flow of the protocol.

In the protocol, the service provider's server first logs on the customer's middleware agent and sends the message [$Account_C$, ID_{MRD}, N_P, $H(SK, M_1)$] to the agent A. Other steps are same as the Step 3, 4, 5 of the first e-reporting and billing protocol.

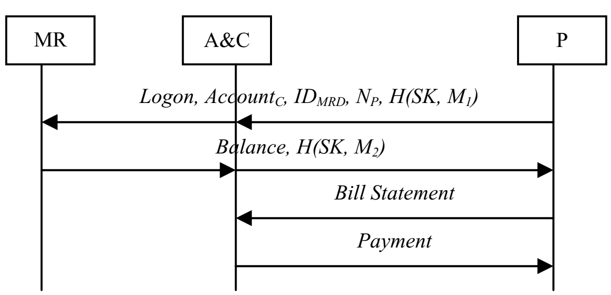


Fig. 7. The Second E-transaction Protocol

5 Architecture Characteristics

5.1 Lightweight Secure Architecture

In order to make the MRD implementation simple and low cost, we only use a keyed hash algorithm in MRD. However, this could provide enough security protection for the system since active attacks (e.g., replay and modify), especially inside active attacks in the home network, are the most important attacks for the reporting and billing system. Furthermore, in order to protect the communications between the agent and server against outside attacks, we employ the existing SSL secure technology. Except for the simple secure algorithm - hash function, we don't use any other complex secure functions like signature in the system server-side application layer. All these make the system implementation simple and low cost.

5.2 Highly Efficient Processing

The secure architecture can automatically report the customers' energy consumption data, generate an e-bill statement and process an e-payment. This would make the system more efficient, and overcome some labor-intensive activities (e.g. physically checking the meters). In addition, the lightweight secure architecture also improves the system scalability.

5.3 Flexible Alternative Implementations

As we mentioned, we have two different implementations for the system which depend on which part (the middleware agent and the service provider's server) initiating the e-reporting and billing protocol. For the former, the advantages include easy implementation, low cost, and convenient customer-side control. The main limitation is the scalability since all customers may initiate their e-reporting protocol in the same time. In order to solve this problem, the server could set up the different initiating time period for the different customers when the customers register their service accounts. For the latter, the advantages include more scalability and convenient server-side control. The main limitation is that the middleware agent needs to be installed in a computer or implemented with a special hardware in the customer side but this would increase the cost. In addition, the extra security protection may need to be considered for the computer and hardware.

6 Security Analysis

In this section, we briefly demonstrate that the secure architecture does provide sufficient protection for the system against active and passive attacks.

6.1 Active Attacks

The new secure architecture provides protection against inside and outside active attacks, for instance, replay and modification. First, the MRD and the server can easily discover the modified or impersonated message by verifying the keyed hashing result since all messages between the MRD and the server have been hashed with a keyed hash function (e.g. $H(SK,M)$), and only the MRD and the server know the secret key (*SK*). Even the customers themselves cannot change the message since they don't have the secret key. Secondly, the server can easily find the replayed message by comparing the Nonce since all messages are sent to MRD from the server with the Nonce (N_P). This is very important since SSL already provides anti-reply protection but the protection cannot reach to MRD, i.e., the anti-reply protection only protects the homeowner not MRD. Finally, for the communications between the agent and the server, they are protected with the SSL technology which includes confidentiality, authentication, and integrity protection.

6.2 Passive Attacks

In the new architecture, as we mentioned in the above, all communications between the agent and the server are protected with the SSL Technology. Outside attackers cannot understand the content of the messages since they are encrypted for the confidentiality protection. It is unnecessary to provide the confidentiality protection between MRD and the agent since the customer has the whole control for his/her home network and the content of messages in the protocols shouldn't be confidential for the customer. Furthermore, the server could very easily verify the attacks even if the hackers successfully attack the home network and change the reporting data.

7 Conclusions

We have presented a secure architecture for the e-reporting and billing of home energy services. Our secure architecture combines lightweight secure technology with SSL technology and provides good secure protection for the system. Our low cost architecture can improve the transaction processing of home energy services substituting electronic processing for labor-intensive manual paper-based work. In addition, we proposed two e-reporting and billing protocols with different implementations and briefly discussed their advantages and limitations.

The secure architecture is based on the existing home energy management system [3], i.e. the local net interface, adapters, and network connections are already existed for the home energy appliances. Of course, it may cost lots if the homeowners don't have this kind of systems in their home.

Acknowledgements

We would like to thank all members of IIT at the NRC of Canada for their support towards our R&D projects in Information Security and Privacy Protection. We are also grateful to the anonymous referees for helpful comments.

References

[1] M. Inoue, K. Uemura, Y. Minagawa, M. Esaki, and Y. Honda, "A Home Automation System", *IEEE Trans. On Consumer Electronics*, CE-31, No. 3, 1985.

[2] D. S. Kim, G. Y. Cho, W. H. Kwon, Y. I. Kwan, and Y. H. Kim, "Home Network Message Specificaiton for White Goods and Its Applications", *IEEE Trans. On Consumer Electronics*, Vol. 48, No. 1, pp. 1-9, Feb. 2002.

[3] M. Inoue, T. Higuma, Y. Ito, N. Kushiro, and H. Kubota, "Network Architecture for Home Energy Management System", *IEEE Trans. On Consumer Electronics*, Vol. 49, No. 3, pp.606-613, Aug. 2003.

[4] SSL 3.0 Specification. http://wp.netscape.com/eng/ssl3/. 1996.

[5] The TLS Protocol Version 1.0. RFC 2246 (Proposed Standard). ftp://ftp.rfc-editor.org/in-notes/rfc2246.txt. 1999.

An Agent-Based on-line Monitoring and Diagnosis System for Distributed Databases

Ching-Fang Hsu, Kuan-Cheng Lin and Roger Lin*

Department of Information Management
Northern Taiwan Institute of Science and Technology
Peitou, Taipei, Taiwan 112, R. O. C.
cfshu@ntist.edu.tw
kuanchenglin@ntist.edu.tw

*LeaLea Technology
Taipei, Taiwan, R. O. C.
rogerlin@llt.com.tw

Abstract:
This paper studies the developing and implementation of an on-line monitoring and diagnosis system for resources management agents for the database management system, such as Oracle 8i and its above versions.

Keywords: Resource Management, Expert System, Agent

1. Introduction

Usually, the Resources Management of an Information System can be categorized into three main areas:

1. Operating System Resources Management: such as Processor、Memory、Disk I/O、Process、logs、and etc.,
2. Network Resources Management: such as network traffic monitoring and analysis,
3. Database Resources Management: such as database administration and performance.

In this paper, we will focus the area of operating system resources management. We will develop and implement an agent-based on-line monitoring and diagnosis system for he database management system, such as Oracle 8i and its above versions. The web-based and agent-based on-line monitoring and diagnosis system is constructed on a Tomcat 4.1.27 web server in the Linux operating system. The system uses the JSP program with the Java Database Connectivity (JDBC) driver to enable access to a database. The JavaServer Pages (JSP) technology has been designed to provide a simplified, fast means of creating web pages that display dynamically-generated content [1].

Section 2 describes the architecture of the agent-based on-line monitoring and diagnosis system. Section 3 describes the data format and report results. Finally, section 4 concludes.

2. System Architecture

The architecture of the proposed system includes three 3 layers: Instance Layer, Schema Layer and Security Layer.

2.1 Instance Layer

The Instance layer includes three functions:

(1). Configuration

(2). Session

(3). SGA

2.2 Schema Layer

The Schema layer includes four functions:

(1). Table space

(2). Table

(3). Indexes

(4). Views

2.3 Security Layer

The Schema layer includes four functions:

(1). User

(2). Role

(3). Profile

3. Data Format and Report

In this section, we define the data format and result report of the proposed system.

3.1 Data format

As the function described in section 2, we define the data format and present some examples as follows.

Instance Layer – Configuration

Parameters	
Field	Example
NAME	user_dump_dest
VALUE	C:\Oracle\admin\test5\udump
ISDEFAULT	FALSE
ISSYS_MODIFIABLE	IMMEDIATE
DESCRIPTION	User process dump directory

Configuration	
Field	Example
STATUS	OPEN
HOST_NAME	JACOB
NAME	TEST5
VERSION	8.1.6.0.0
INSTANCE_NAME	test5
STARTUP_TIME	2005-06-07 19:17:19.0
OPEN_RESETLOGS	NOT ALLOWED
LOG_MODE	NOARCHIVELOG
OPEN_MODE	READ WRITE

Instance Layer - Sessions

Sessions	
Field	Example
SID	8
CONSUMED CPU TIME	0
PHYSICAL READS	13
STATUS	INACTIVE
USER NAME	DBSNMP
OS USER	SYSTEM
PROCESS	1096:1280
MACHINE	WORKGROUP\JACOB
PROGRAM	dbsnmp.exe
RESOURCE CONSUMER GROUP	DEFAULT_CONSUMER_GROUP

Instance Layer – SGA

SGA	
Field	Exmple
shared_pool_size	4,920,320
Buffer_Cache	16,777,216
large_pool_size	614,400
Jjava_pool_size	32,768
Total SGA	22,344,704
sort_area_size	65,536

Schema Layer – Indexs

Indexs	
Field	Exmple
INDEX_NAME	AQ$_MSGTYPES_PRIMARY
OWNER	SYS
TABLE_NAME	AQ$_MESSAGE_TYPES
TABLE_TYPE	TABLE
PARTITIONED	NO
STATUS	VALID

Schema Layer – Tablespaces

Tablespaces	
Field	Exmple
TABLESPACE_NAME	SYSTEM
FILE_NAME	C:\ORACLE\ORADATA\ TEST5\SYSTEM01.DBF
CONTENTS	PERMANENT
EXTENT_MANAGEMENT	DICTIONARY
(Byte)ISSYS_MODIFIABLE	60,817,408
(Byte)Used	53,739,520
(%)Used	88.36207

Schema Layer – Views

Views	
Field	Exmple
OBJECT_NAME	ALL_ALL_TABLES
STATUS	VALID

Schema Layer - Tables

Tables	
Field	Exmple
OWNER	SYS
TABLE_NAME	AQ$_PENDING_MESSAGES
PARTITIONED	NO
NUM_ROWS	1,235
LAST_ANALYZED	2005-06-07 19:17:19.0

Security Layer – Users

Users	
Field	Example
USERNAME	SYS
ACCOUNT_STATUS	OPEN
EXPIRY_DATE	2005-04-30
DEFAULT_TABLESPACE	SYSTEM
TEMPORARY_TABLESPACE	TEMP
PROFILE	DEFAULT
CREATED	2005-04-30 05:20:14.0

Users Role Privilege	
Field	Exmple
Type	ROLE
Privilege	CONNECT
Owner	DBSNMP
Name	Name
Grant/Admin	NO
Grantor	SYSTEM

Security Layer – Roles

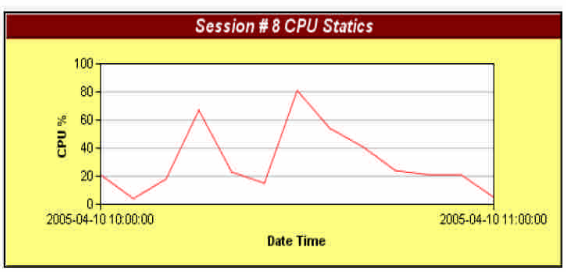
Roles	
Field	Exmple
Role	AQ_ADMINISTRATOR_ROLE
Privilege	EXECUTE
Schema	SYS
Object	DBMS_RULE_EXIMP
Grant/Admin	NO

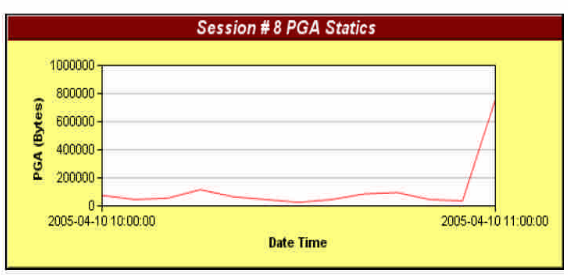
Security Layer – Profiles

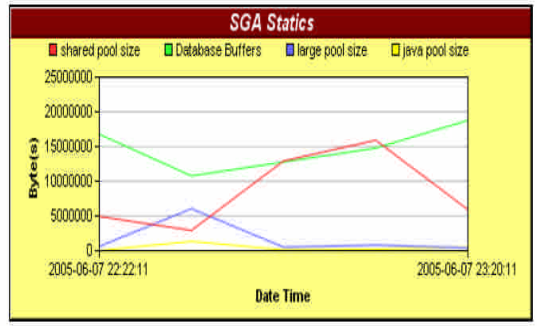
Profiles	
Field	Exmple
PROFILE	DEFAULT
RESOURCE_NAME	COMPOSITE_LIMIT
RESOURCE_TYPE	KERNEL
LIMIT	UNLIMITED

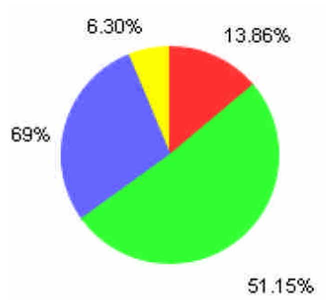
3.2 Result Report

The four result report examples are displayed as follows.









4. Conclusions

This paper studies the developing and implementation of an on-line monitoring and diagnosis system for resources management agents for the database management system, such as Oracle 8i and its above versions.

The architecture of the proposed system includes three 3 layers: Instance Layer, Schema Layer and Security Layer.

REFERENCES

[1] Java Server Pages (**JSP**) Specifications http://java.sun.com/**jsp**

An Agent-Based on-line Monitoring and Diagnosis System for Heterogeneous Servers

Ching-Fang Hsu, Kuan-Cheng Lin and Roger Lin*

Department of Information Management
Northern Taiwan Institute of Science and Technology
Peitou, Taipei, Taiwan 112, R. O. C.
cfshu@ntist.edu.tw
kuanchenglin@ntist.edu.tw

*LeaLea Technology
Taipei, Taiwan, R. O. C.
rogerlin@llt.com.tw

Abstract:

In this study, we discuss the implementation of a resources monitoring agent for popular operating system , such as Windows 2000 server、SUN Micro Solaris 2.7, Linux and its above versions. This agent system will implement five monitoring functions: System Resources Monitoring, Error Message Monitoring, Event Control, Auditing System and System Health Checkup.

Keywords: Resource Management, Monitoring System, Agent

1. Introduction

For the past two decades, the distributed、open computer systems has replaced the proprietary and closed computer system such as IBM Mainframe. The distributed、diverse computer systems do make the resources management of information system become more difficult. System administrator in an enterprise cannot manage the sporadic and diverse computer system effectively and cannot even know timely when failures occur to the system. Therefore a central controlled resources monitoring and management system could be very helpful to system administrator.[1]

Usually, the Resources Management of an Information System can be categorized into three main areas :

1. Operating System Resources Management: such as Processor、Memory、Disk I/O、Process、logs、and etc.,
2. Network Resources Management: such as network traffic monitoring and analysis,
3. Database Resources Management: such as database administration and performance.

In this paper, we will focus the area of operating system resources management. We will develop and implement an agent-based on-line monitoring and diagnosis system for heterogeneous servers with different operating system such as Window2000、SUN Micro Solaris 2.7

and its above versions.

Section 2 describes the function of the agent-based on-line monitoring and diagnosis system. Section 3 describes the implementing issues of the agent-based on-line monitoring and diagnosis system. Finally, section 4 concludes.

2. System Function

This system will build resources monitoring agent for popular operating system, such as Linux, Window2000、SUN Micro Solaris 2.7, and its above versions.

This system will implement five monitoring functions:
(1). System Resources Monitoring：includes CPU、Memory、File system、Process、User utilization etc.

(2). Error Message Monitoring: is to track critical system message and issue an event in time to notify system administrator to take action before a real disaster occurs.

(3). Event Control: can collect the event happened on the different servers and design the corresponding actions.

(4)Auditing System: can help system administrator to monitor all the user behavior on the individual system, for example it can audit

a). the history of all commands has been executed on the system,

b). the successful and failed login history,

c). the record of password changed,

d). the record of switch user,

e). the history of failure of file access.

(5). System Health Checkup: will define some very important daily system checkup procedures that could help administrator to collect all security checkups every day automatically that could lower the administrator's burden to some degree.

All the information collected by agent will be sent to the monitoring server and to be processed by the Management Server. And then, the processed information be displayed on the console. These data can be analyzed further to become useful data for decision maker that will have a clear picture about how the system has been used and have future plan for upgrading their system accurately.

3. Issues of Implementations

In this section, we will describe the implementing issues of the agent-based on-line monitoring and diagnosis system. It includes system architecture and agent design.

3-1 System Architecture

The system architecture is depicted as Fig1.

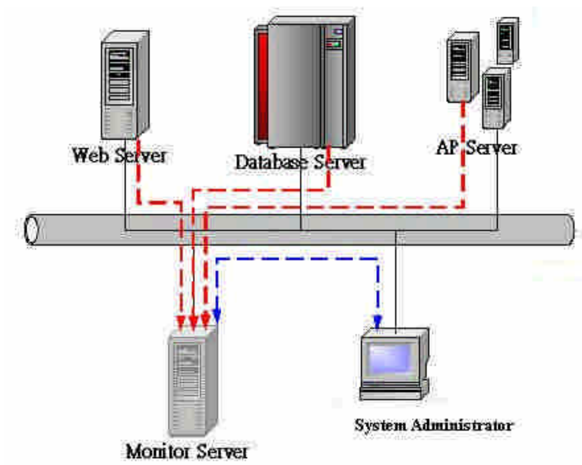


Figure 1: The System Architecture of the agent-based on-line monitoring and diagnosis system.

The web-based and agent-based on-line monitoring and diagnosis system is constructed on a Tomcat 4.1.27 web server in the Linux operating system. The system uses the JSP program with the Java Database Connectivity (JDBC) driver to enable access to a database. The JavaServer Pages (JSP) technology has been designed to provide a simplified, fast means of creating web pages that display dynamically-generated content [3].

We build a resource management system (management server) to collect various data from different client agent, residing on various computers with different operating system. It will store the data as well as provide other data-processing features such as "display"、"event notification"、"reporting" etc.

3-2 Agent Design

The agent is designed and constructed on heterogeneous server with different operating system.

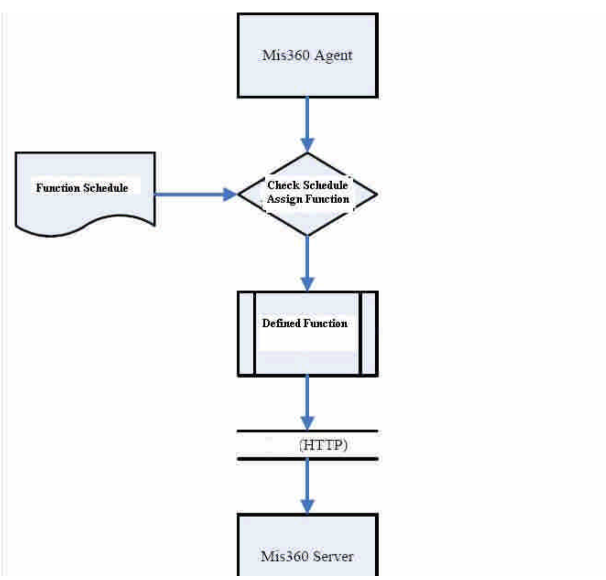


Figure 2: The agent function of the agent-based on-line monitoring and diagnosis system.

In Linux and SUN Micro Solaris 2.7 and its above versions operating system, the agent is implemented by using java and script language [2]. And, in the server with Windows 2000 server operating system, the agent is implemented by using .Net language [4].

The agent will monitor the event occurred on the server and send the data to the management server by using the schedule and designed function.

4. Conclusions

In this study, we build resources monitoring agent for popular operating system , such as Linux, Window2000、SUN Micro Solaris 2.7 and its above versions. And, the system includes five monitoring functions: System Resources Monitoring, Error Message Monitoring, Event Control, Auditing System and System Health Checkup.

All the information collected by agent will be sent to the monitoring server and to be processed by the Management Server. And then, the processed information displayed on the console. These data can be analyzed further to become useful data for decision maker that will have a clear picture about how the system has been used and have future plan for upgrading their system accurately.

REFERENCES

[1] Z. Cui, B. Odgers, and M. Schroeder, "An In-Service Agent Monitoring and Analysis System," Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence, pp.237-244, 1999.

[2] D. B. Lange, M. Oshima, Programming and Deploying Java Mobil Agents with Aglets,

Addison-Wesley, ISBN 0-201-32582-9, 2nd printing, November 1998.

[3] Java Server Pages (**JSP**) Specifications http://java.sun.com/**jsp**

[4] http://www.microsoft.com/net/

Application of mobile agents in wireless-based mission critical emergency operations

Hamidreza Pousti
Department of Information Systems
University of Wollongong, Australia

Helen Hasan
Department of Information Systems
University of Wollongong, Australia

Eryk Dutkiewicz
Wireless Technology Laboratory
University of Wollongong, Australia

Abstract

In the ambiguous, high stress, and time sensitive environment of mission critical emergency operations, the decision-making process could benefit from a distributed pattern of mobile resources across a wireless network. Dealing with challenges in these environments, mobile agents can move across the wireless setup, having many promising attributes including autonomy, adaptability, and persistency. These attributes make them able to act independently, to sense and interact with their environment and to make decisions on behalf of the users. Mobile agents are particularly suitable for the situations where the distributed information throughout a network is crucial for other applications/services/agents which heavily rely on, and are sensitive to situation-based and well-timed information. Until now there has been a lack of studies that approach issues of disaster response in a way that integrates these technologies while also considering the human and organisational perspectives. In this paper we present and investigate an intense man-made disaster. Results show that mobile agents can in many ways augment the performance of communication systems working in extreme environments of mission critical emergency operations.

1. Introduction

The news in recent years has been full of natural disasters, terrorist/criminal attacks, conflicts and accidents, many of which occur suddenly, are unexpected and often unanticipated and result in untold damage, suffering and grief. Whether at a local, national or international level, these events require a rapid response involving the instant co-ordination of people, teams and equipment from many different organisations. These include various police forces, fire brigades, medical personnel, government and non-government agencies and so on. While the emergency response organisations are trained and prepare for such contingencies, every catastrophe has its own unique set of circumstances and conditions that challenge the response. In these environments, the rescue operations are can be acknowledged as mission critical emergency operations (MCEO) which are characterized by ambiguity, high stress, and time dependency of decision-making where decisions on appropriate courses of action have to be made under stress, in real time and often with incomplete and conflicting information. In these situations, the decision-making process follows a distributed pattern in which mobile team members with different skills are distributed across a networked operational environment. In fact, human rescuers working in critical emergency operations must communicate to each other in order to make quick decisions under stress and get victims to safety often at the price of risking their own lives. One of their main difficulties is that communications may be intermittent or non-existent, hence partial and incomplete information becomes a norm of environment and it is not an exception any more.

Following a major emergency involving threats to public safety, extensive enquiries are conducted through the lessons learnt. This results in improvements both to technologies that can assist in these situations and in knowledge on how people make decisions and act. This study aims to add to the understanding of the technologies, human issues and other socio-technical systems formed in emergency situations.

The purpose of this paper is to discuss the contribution of mobile agents - as a new paradigm of information gathering, processing and dissemination - together with new wireless networks to support mission critical emergency operations. The next section of this paper will discuss the related issues and concepts about software agent technology, mobile agents and their potential applications in MCEO. Section 3, introduces a secondary case study and discusses how mobile agents can contribute to the MCEOs' better performance and success. In the section 4 we draw our conclusions and suggest areas for future studies.

2. Background Issues and Concepts

Previous studies [1], [2], [3] show that one major challenge of emergency first respondents to a large-scale disaster situation is the relevant information that may not get accessed by the rescue workers whose operations and life depend on that information. The problem is profound when the distribution of information is slow enough to become obsolete and it cannot reflect the real environmental situations for the users who must respond properly to the changes of dynamic environment. So, it seems that existence of technologies by which first responders will be able to make more accurate and timely decisions is indispensable.

Using the concepts of Mobile Ad hoc Networks (MANET) technologies [4] and situations involves in MCEOs, they have inspired our research to stand for an autonomous collection of mobile users that communicate over relatively bandwidth-constrained wireless links. Since the nodes are mobile, the network topology may change rapidly and unpredictably over time. Its challenges are:

- Since the nodes are mobile, the network topology may change rapidly and unpredictably over time.
- The network is decentralised; all network activity must be executed by the nodes themselves.
- Need efficient distributed algorithms to determine network organisation, link scheduling, and routing.
- Shortest path is not the best path.
- The network should be able to adaptively alter the routing paths to alleviate any of these effects.
- In many environments, preservation of security, latency, reliability, intentional jamming, and recovery from failure are significant concerns.

Table 1. MANET topologies
Suitability for dynamic environments

Topology	Reliability	Adaptability	Scalability
Point to Point	High	Low	None (two end points)
Point-to-Multipoint	Low	Low	Moderate (7-30 endpoints)
Mesh Networks	High	High	Yes (thousands of endpoints)

As indicated in Table 1, in the preferred mesh topology a node can send and receive messages and can function as a router and can relay messages for its neighbours with:

- Self-Configuring and Self-Healing
- Redundancy and Scalability
- Easy installation in short period of time
- No requirement for sophisticated planning and site mapping to achieve reliability

If environmental conditions result in poor reliability, it is difficult or impossible to adapt a point-to-point or point-to-multipoint network to increase reliability. By contrast, mesh networks are inherently reliable, adapt easily to environmental or architectural constraints, and can scale to handle thousands of end points.

Among possible wireless technologies one candidate is the Ultra-wideband (UWB) radio that can provide a reliable communication system by supporting multi-services such as voice and video communications and biometric information [5]. Its specific characteristics include:

- Operating across a wide range of frequency spectrums by transmitting a series of very narrow and low power pulses.
- Being immunes to the multipath cancellation effects as observed in mobile and in-building environments.
- Having a low energy density which translates into a low probability of detection
- Combining well with the mesh network topology [6]

Moreover, the technology of software agents has emerged as a suitable metaphor for representing the computer processes that assist human decision-making. Such software agents should not only retrieve information on request; but they should actively and intelligently anticipate, adapt and actively seek ways to support users [7]. In addition, they can reduce the amount of interaction between humans and the computer system and allow humans to concentrate on other activities, such as assessing the situation, making decisions, or reacting to changes in the system [8]. This timesaving is critical in the MCEO. Other research [9],[10] reveals that mobile software agents feature properties that should prove a helpful component for integrated responses to civilian and military crises in unpredictable, time sensitive environments. Mobile agents have many promising attributes which make them able to act independently, sense and interact with their environment and make decisions on behalf of the users.

Until now there has been a lack of studies that approach issues of disaster response in a way that integrates these technologies and also consider the human and organisational perspectives. There are some studies on the new wireless technologies and some research about mobile agents but there are no reports about the combination of these two technologies in the context of emergency response tasks, addressing the advantages and challenges of coupling the capabilities of mobile agents with new wireless technologies.

The following section will provide a brief discussion about the mobile agents and their attributes.

3. Mobile agents and MCEOs

Agents are inspired by many diverse disciplines and draw on many different areas of research, such as computer science, information systems, psychology, engineering and so forth. However, it is difficult to find a succinct definition for an agent that includes all the aspects that most researchers and developers consider an agent to be, and exclude all of the aspects they are not considered to be. In General, an agent is anything that can be viewed as perceiving its environment through the sensors, and acting upon that environment through actuators [11]. Therefore, an agent can be a human, a robot, a piece of software or whatever else that could interact with its environment by sensing and acting upon its perception of the environment and related conditions.

Agents generally may exhibit different attributes. These attributes include: autonomy (the ability of operate without the direct intervention of other agents, making independent decisions, and having some kind of control over their actions and internal states), reactivity, collaborative behaviour, persistency, adaptivity and learning, pro-activeness and, mobility (the ability to migrate in a self-directed way from one host platform to another one).

The term 'mobile agent' contains two distinct concepts of mobility and agency and for software agents, it refers simply to a self-contained and identifiable computer program that can move within the network and act on behalf of the user or another entity. This is in contrast with stationary agents that can communicate with their environment through traditional means, such as remote procedure calling and messaging and they can execute only on the system on which they start execution. According to Macaire et al. [12] the ability of a mobile agent to transport itself from one system to another, allows it to move closer to a system that contains an object with which it wants to interact. This characteristic of mobile agents, makes them qualify to operate in volatile environments as a new communication paradigm. In this new communication model, according to White [13], mobile agents are able to communication between hosts where hosts not only call procedures in one another but also supply procedures to be performed there. As the result the ongoing interaction does not require ongoing communication and this will have a considerable impact on conserving the network bandwidth particularly when the bandwidth is a scarce resource. Accordingly, mobile agents provide an alternative method to the remote procedure calling (RPC) communication paradigm where communications require bandwidth hungry and ongoing interactions between distant hosts. This would be possible by the ability of mobile agents to store important data about themselves and their environment in order to remain flexible and responsive to their environment.

Every mobile agent embraces different categories of data which are essential to accomplish a particular or series of tasks. Liu [14] describes those data by portraying mobile agents as 'serializable objects' whose data as well as their state can be marshalled for transportation over the network. These essential data can basically be divided to four groups: identification data, itinerary and routing data, task specifications data, and the data concern the logic code to perform its tasks. In this approach, mobile agents travel from one host to another and make use of local resources and the data they carry with themselves to perform their tasks and meet their objectives.

Although there is no killer application for mobile agents [10], mobile agents have shown great potential to work and contribute in different areas. Generally, in a typical wireless network, disconnections may happen particularly when the operation of the network is dynamic and the environment is extreme. In such an environment, mobile agents have shown great potential to handle valuable information, even when the connection has failed.

This application of mobile agents is of great interest in the teamwork activities and tasks such as healthcare, military and emergency first response situations. We categorize these activities as Mission Critical Emergency operations (MCEO) where the environments are extreme and decisions makers every level require precise and timely information about their environment. Studies by McGrath et al. [15] show that the co-existence of mobile agents and wireless radio technologies can provide a reliable framework, by which many problems in the current communication systems in military and public safety operations can be addressed. In fulfilling important requirements of collaborative teams involved in an extreme environment operation, mobile agents can play crucial roles in supplying a robust infrastructure for disseminating vital information across unreliable networks. Moreover, other research reveals that mobile agents have a considerable impact on decreasing the required time for extracting sensitive information that supports decision makers in time critical tasks such as military and emergency first responders operations. Payne et al. [3] argue that in time critical tasks, mobile agents have shown potential in supporting both individuals and teams in terms of reducing the required time to make a proper decision and cope with uncertainties. This is also described by Hofman et al. [16] as the means by which application of mobile agents can improve the efficiency of military or civilian tactical operations in unreliable, low bandwidth networks.

4. Research Approach

This research describes the application of the above mentioned concepts to a secondary case study, in order to advance the understanding of their integration in practice. This case is analysed interpretively. A case method is considered applicable to this type of research, which attempts to integrate diverse paradigms as it provides a rich yet pragmatic approach to data collection and analysis. The use of a secondary case is necessary for this topic since: 1- it provides a real scenario, which is not possible to repeat or access first hand since, by their nature, emergency situations are not planned. 2- It would, in any case, be unethical to collect data during an event when all resources are needed to cope with the problem. Most big events are well researched through exhaustive data collection after the event and most reports are in the public domain as it is a public issue. According to Reason [17] when sufficient evidence regarding a single case is available then "we are able to study the interaction of the various causal factors over an extended time scale in a way that would be difficult to achieve by other means". While it is not suggested that it is possible to generalise the findings of a single case, as each is unique to its own context, there are common threads that enable learning to occur across cases. Walsham [18] contends that interpretive case methods are appropriate when the aim of research is to understand the context and the process of systems, rather than to establish any hypothesis for testing.

5. Case Study

5.1 Background to Case Study

The application of wireless technology and mobile agents to the public safety domain and rescue operations was evaluated by examining, a portion of the September 11 of 2001 World Trading Centre collapse (WTC). Background knowledge of the WTC collapse can provide insight into the communication processes, complex interactions, and the accident sequences itself. Respectively, it would help to understand better how radical the environment was, from the time zero (first hit) until the last moment that the second tower collapsed. Furthermore it can provide a better insight to the crucial role of communications systems which could lead the decision makers to make appropriate decisions in a timely fashion based upon different kinds of data they receive from different of sources. Also studying the WTC collapse is significant since the size of the event is huge. According to the centre of fire statistics of International Technical Committee for the Prevention and Extinction of Fire [19], in the September 11, 2001 the biggest number of firefighters' fatalities in the history of civilian rescue operation has occurred.

So far much research has been carried out regarding the September 11, 2001 WTC collapse and it has been studied from different points of view. However, to the best of our knowledge, the application of new wireless technologies and mobile agents has not been investigated in this context. Considering the unique significance of this case, this study could provide a better understanding of new applications, implications, advantages and disadvantages of these technologies, especially in the situations of extreme environment rescue operations.

5.2 Review of the Case

On the September 11, 2001 at 8.46am the first plane hit the North tower of WTC. An American Airline Boeing 767, was deliberately flown to the North tower, striking it across 93 to 98 floors. The aircraft was swallowed up by the building as it hit to the building in 440 mile per hour and had a massive impact on the North tower since it was running too fast and also it was too heavy (Figure 1).

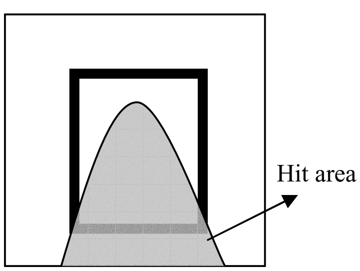


Figure 1: Hit area of North Tower [20]

For the 6000 people below where the plane had hit, the staircases still offered the means of escape but for the 950 people caught above the point of impact on the fire, there was no way out.

At 9.02am the second airplane hit the South tower, impacting between floors 78 to 84. The plane had sliced into the South tower at an angle to the right and thereby smashed the eastern wall (Figure 2).

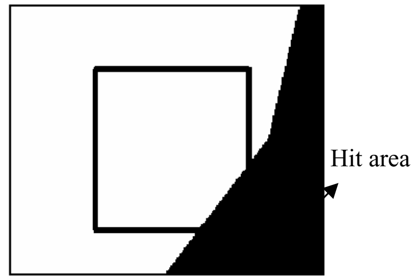


Figure 2: The area of South Tower [20]

At this stage about 2000 people left alive in the South tower, some 1500 below where the plane had crashed. For the 500 left above the impact line, two of the three staircases were completely impossible and one still offered the way out in its minimum capacity due to the heavy smoke and fire.

At 9:51 a.m., a helicopter pilot cautioned that "large pieces" of the South Tower appeared to be about to fall and could pose a danger to those below. Immediately after the tower's collapse, a helicopter pilot radioed that news. This transmission was followed by successive communications at 10:08 a.m., 10:15 a.m., and 10:22 a.m. that called into question the condition of the North Tower. The FDNY chiefs would have benefited greatly if they had been able to communicate with personnel in the helicopter.

57 minutes after the second attack, At 9.59 a.m. the South Tower collapsed. At 10.03 a.m., four minutes after the first tower collapse, an urgent evacuation message was radioed to all firefighters and policemen in the North tower. Many FDNY personnel in the North Tower who received the evacuation orders did not respond uniformly for several reasons. 25 minutes later at 10.28 a.m. still nearly 1000 people were trapped in the upper 20 floors, plus many firemen unaware of the evacuation order and still inside the tower were trying to save people, the North Tower came down. In total 2800 people died in the attack, 479 of them were from the emergency services personnel, including 343 firefighters.

5.3 Case Study Analysis

After the first tower was hit by the hijacked airplane many firefighters, police officers and other rescue units got to the scene as soon as they could. But unpredictable problems arose causing obstacles to cope with the huge magnitude of the disaster properly. Any attempt to establish a unified command would have been further frustrated by the lack of communication and coordination among responding agencies. The command posts were in different locations which could have served as a focal point for information sharing and they did not play an integrating role in ensuring that information was shared among agencies. There was also a lack of comprehensive coordination between rescue units (Fire Department of New York: FDNY, New York Police Department: NYPD, and Port Authority of Police Department: PAPD) personnel climbing above the ground floors in the Twin Towers.

Our studies show that on September 11 2001, information that was critical to informed decision making and required actions, was not shared among the involved agencies mainly for the following problems:

Large number of users involved in the radio communication: this issue had affected the reliable one-to-many broadcast capability of the firefighters' communications systems due to their communication systems limitations.

Lack of interoperability among different parties: One considerable issue in the, WTC fall was the interoperability problem and there was insufficient interoperability among different communication systems.

Data traffic and network congestion: communication networks experienced a number of technical problems. One important challenge that undermined their performance was the data congestion and high network traffic throughout the networks. As the case study shows, this problem highly damaged the communication reliability, coverage area and networks' responsiveness.

Lack of Meta Data: Passing of information from one person to another, with understanding, is the fundamental purpose of any communications system and interpersonal communications normally include more than just verbal exchanges. As Marstrander & Hanssen [2] state, emergency communicators usually don't have the luxury of using all of these elements that make up interpersonal communications. Their communications are limited to what is spoken or heard and a well trained emergency communicator and appropriate listening skills are critical for successful communications.

Network interruptions and volatility: One important issue encountered through the case study is the frequent network breakages because the network was overwhelmed by the users or due to some technical problem such as communications infrastructure damages.

Intelligent communication systems with autonomy: Public safety communications through the radio systems rely primarily on voice communication and they are designed to convey voice in the best way possible. But these systems do not have any features that make them capable of sensing their environment and perform some processes and decision making based upon the perceived situation. Essential for these systems is that they should be authorized and capable to act autonomously. Also they should be able to establish a relation with their environment and perform within their environment proactively and reactively. This capability can be translated to the degree of systems' intelligence in order to integrate and transfer many of the human tasks to the systems functions.

5.4 Discussions

The case of the WTC collapse and the results show that the challenges of providing high quality communications to computing systems and users in a mobile environment is not only because the communications medium is wireless, but rather that the system must deliver information in the face of a constantly changing environment. According to Katz [21] in these kinds of environments the communications systems should be 'adaptive' since mobility requires adaptability. Systems must be location and situation-aware, and must take advantage of the information to dynamically configure themselves in a distributed fashion. These systems require flexible access, in real-time, to multimedia information sources through wireless information systems such as: support for decision makers in the field, crisis management and response, law enforcement, and so forth. To be adaptable, communication systems require sensing their execution environment and then reacting accordingly to changes. These reactions in turn, should be autonomous to assure that the taken actions are independent of any other factors that might affect the adaptability of the systems. In this sense, adaptability of a communication system depends on its sensitivity, reactivity, pro-activity, and autonomy a mobile configuration. Based upon this idea, mobile agents can make considerable contributions in communication systems, making them flexible and adoptable to their environment. As discussed in previous section, mobile agents are autonomous agents that can move throughout the networks and sense their environments through sensors and then choose the best action based upon the known conditions. In fact mobile agents can help communication systems not only augment the voice communications but also they are able to bring many benefits to them, make them flexible and compatible to the changing environments. Some of these benefits are described by Lange & Oshima [10] as: 1- they reduce the network load, 2- they can overcome network latency, 3- they can encapsulate network protocols, 4- they execute asynchronously and autonomously, 5- they are able to adapt dynamically to their environment, 6- they are naturally heterogeneous, and 7- they are robust and fault-tolerant. These benefits of mobile agents along with their other attributes such as persistency, ability to learn, and ability to cooperate with other agents make them an appropriate option to be exploited in communications systems. In fact these contributions and effects can augment the influences and effectiveness of voice communications by providing complementary information about the environment such as retrieving data from databases [9] and handling different formats of data such as pictures and maps [22]. This makes mobile agent-based communication systems able to cope with large numbers of users involved in communications, deal with the interoperability problem, reduce the data traffic and network congestion, provide and process additional data about the environment, and work in a volatile network without losing the process ability. This is quite remarkable for decision makers when they are involved in a time sensitive and mission critical emergency task.

6. Conclusion

Mission critical emergency tasks require communications systems, that effectively support their operation and provide them with proper quantity and quality of information in a timely fashion. The case study presented in this paper reveals to what extent the role of communication systems is important in success or failure of MCEOs. In this paper, we investigated the contribution of mobile agents as a new communication paradigm in the MCEOs. Studies show that many attributes of mobile agents can augment performance of communication systems working in extreme environments. These attributes and characteristics of mobile agents make them suitable to cope with intermittent wireless networks, process data at its location and provide users with more reliable and meaningful information. These are vital for emergency tasks operations since they can equip mobile users with more information about their situation, save their time, speed up their operation and save more lives with less risk.

REFERENCE:

[1] Commtech 2003, *Can't We Talk? Interoperability, Working Together To Bridge the Communications Gap To Save Lives A Guide for Public Officials*, National Task Force Publication under the Agile Project Programme, , USA.

[2] Marstrande, A. & Hanssen, I. B. 2002, *Fulfilment of Requirements in Public Safety Wireless Communications using TETRA and GSM-R*, Norwegian University of Science and Technology (NTNU), Department of Telecommunications.

[3] Payne et al., 2000, 'Agent-Based Team Aiding in a Time Critical Task', *Proceedings of the 33rd Hawaii International Conference on System Sciences.* Hawaii. USA.

[4] Lewis, N. 2004, 'Wireless Sensor Networks', in Cook, D.J. and Das, S. K. (eds.), *Smart Environments: Technologies, Protocols, and Applications*, John Wiley, New York.

[5] Fontana, R. et al., 2002,' Recent Advances in Ultra Wideband Communications Systems', *Proceeding of the IEEE Conference on Ultra* Wideband Systems and Technologies, Baltimore, MD.

[6] Poor and Hodges 2002, *Reliable Wireless Networks for Industrial Systems,* Technical White Paper, Ember Corporation, 2002 [online] Available: http://www.ember.com/resources/whitepapers/index.html [Accessed: 14 October 2004].

[7] Bradshaw, M.J (eds.) 1998, *Software Agents*, AAAI Press, Cambridge, Massachusetts.

[8] Zachary, W., Le Mentec, J-C., and Ryder, J. 1996, 'Interface agents in complex systems, in Ntuen, C. A. and Parks, E. H. (eds.), *Human Interaction with Complex Systems: Conceptual Principles and Design Practice*, Kluwer Academic Publishers Proceedings of the 33rd Hawaii International Conference on System Sciences.

[9] McGrath, S. P., Chacon B., and Whitebread, N. 2000, 'Intelligent Mobile Agents in the Military Domain,' *Proceeding of the Fourth International Conference on Autonomous Workshop on Agents in Industry*, Barcelona, Spain.

[10] Lange, D. & Oshima, M. 1999, Seven Good Reasons for Mobile Agents', *Communications of the ACM*, Vol. 42, no. 3, pp 87-89.

[11] Russell, S. & Norvig, P. 2003, *Artificial intelligence: a modern approach,* Prentice Hall, New Jersey

[12] Macaire, A., Carlier, D., and Paradinas, P. 1999, 'A Personal Agent Manager for Mobile Users',', *Proceeding of the European Multimedia Embedded Systems and Electronic Commerce Conference (EMMSEC'99)*, London.

[13] White, J. 1996, *Mobile Agents White Paper*, General Magic, USA.

[14] Liu, M. L. 2004, *Distributed Computing Concepts and Application*, Addison Wesley, USA.

[15] McGrath, S. P. et al., 2003, *ARTEMIS: A Vision for Remote Triage and Emergency Management Information Integration*, Dartmouth University, [online] Available: http://www.ists.dartmouth.edu/library/other/artemis.pdf [Accessed 16 October 2004].

[16] Hofman, M. O., McGovern, A., and Whitbread, K. R. 1998, 'Mobile Agents on the Digital Battlefield', *Proceeding of Second International Conference on AUTONOMOUS AGENTS (Agents '98)*, Minneapolis, St. Paul.

[17] Reason, James. 1990, *Human Error,* Cambridge University Press: New York.

[18] Walsham, G. 1995 Interpretive case studies in IS Research: nature and method. *Eur. J. Inf Systems* 4: 74-81.

[19] International Technical Committee for the Prevention and Extinction of Fire 2003, *Annual Report 2003*, Moscow. [online] Available: www.CTIF.org [Accessed 10 October 2004].

[20] *The fall of the World Trade Centre* 2002 video recording, BBC, U.K.

[21] Katz, R. H. 1996, 'Adaptation and Mobility in Wireless Information Systems'. *IEEE Personal Communications*, 1(1),

[22] Gray, R.S. 2000, 'Soldiers, Agents and Wireless Networks: A Report on a Military Application', *Proceeding of the PAAM 2000*, Manchester, England.

A Bluetooth-Based Wireless and Parallel Computation Environment for Matrix Multiplication

Gwo-Jong Yu, Chih-Chu Wu, and Cheng-Ku Lai
Department of Computer and Information Science
Aletheia University, Taipei, Taiwan
yugj@email.au.edu.tw

Abstract

In this paper, a wireless and parallel computation environment based on Bluetooth technology is proposed and illustrated by a matrix multiplication problem. The proposed algorithm divides a large computation problem into multiple smaller computation sub-problems. Each sub-problem is assigned to and solved by a mobile device. Individual results are collected through wireless communication and integrated into a complete solution. As demonstrated by experimental results, the proposed approach can reduce computation time, increase performance and provide flexible wireless computation environment.

1. Introduction

As wireless technologies advances rapidly, the number of devices which can perform mobile computation also increases. It is possible in the near future that massive mobile computation paradigm will replace conventional wired and fixed computation paradigm. Among mature wireless technology, Bluetooth technology has advantages of low cost, low power consumption, and ad hoc connection. Bluetooth adopts frequency hopping approach to reduce the effect of interference. Although the computation power is low in current Bluetooth enabled mobile devices, such as PDA and mobile phone, wireless parallel computation can reduce computation time of complex problems. So, it is important to propose a wireless parallel environment, so that the power of mobile computation can be fully explored.

The parallel computation framework proposed in this paper is based on wireless communication technology. The matrix multiplication problem is applied to illustrate the operations of wireless parallel computation framework. Through divide and conquer mechanism, a large matrix computation problem can be recursively divided into multiple small matrix computation sub-problems. A small matrix computation is suitable to be computed in mobile device with low computation power. The computation time can thus be reduced significantly. To apply divide and conquer concept to Bluetooth, a master in piconet is responsible to divide problem into smaller sub-problems and distributed these sub-problems into slaves. Using role switching [1] technique, a slave can change its roles to be a master in another piconet. This master can further divide sub-problems into even smaller problems. The decomposition process forms a fully complete tree structure.

In literature, lots of researchers discuss parallel matrix computation problems. Wang and Sahni [2] proposed matrix multiplication algorithm in TIS-Mesh environment. This algorithm can minimize inter-processors communications, so that the performance of entire system can be increased. However, Wang and Sahni's work can only be applied in wired network. Recently, parallel computation in wireless environment becomes an important research problem. The objective of this paper is to extent the parallel computation mechanism from wired to wireless and mobile environment.

For researches of wireless parallel computation, Wu and Huang [2] proposed a framework for wireless parallel computation. The framework extends parallel computation task from distributed shared memory architecture to wireless architecture. They also proposed algorithm to find maximum value and sort array. In a network with N processors, their algorithm can reduce time complexity to O(1). However, their method did not take bandwidth allocation and scheduling problem into considerations. Thus, collision and inefficient bandwidth scheduling reduce the system performance. On the other hand, Chang *et al.* [3] proposed a BlueCube architecture based on Bluetooth technology. Chang's method constructs a wireless parallel computation environment which is similar to Hyper Cube architecture. Advantages of Chang's algorithm include small average length of routing path and small average number of paths served by each device. Besides, Chang's method also proposed a distributed network migration protocol within a piconet. This protocol decomposes a congested piconet into multiple piconets through role switching techniques. However, Chang's method did not evaluate performance of parallel computation. Based on discussions of previous works, the protocol presented in this paper will focus on efficient bandwidth allocation and communication scheduling algorithms, so that the performance of entire network can be maximized. A matrix multiplication problem will be applied to illustrate the proposed approach and evaluate performance of the approach.

2.The Matrix Multiplication Problem

Standard and Strassen are two well known matrix multiplication algorithms. In the following, we will first introduce Strassen algorithm.

2.1 Assumption and Definition

In this paper, all devices which are participated in parallel computations can communicate with each other. The network is homogenous, so the computation power of each device is the same. For the sake of presentation, mathematical symbols are defined as follows:

$A_{m,n}(i,j)$, matrix: The element in the i-th row and the j-th column of matrix $A_{m,n}$, where m and n are number of rows and columns in matrix A, respectively. When m=n, the maxtix $A_{m,n}$ can be abbreviated as A_m.

$C_{m,n}(i,j)$, product of matrix multiplication: The element of the i-th row and the j-th column of matrix $C_{m,n}$, where $C_{m,n}$ denotes product of matrices $A_{m,p}$ and $B_{p,n}$. The operation of matrix product is denoted as $C_{m,n} = A_{m,p} \times B_{p,n}$

m, size of C_m: In this paper, Strassen algorithm is used to investigate wireless parallel computation. To simply the discussion, it is assumed that m is power of 2, i.e. $m = 2^k$.

$Dt(M)$, Required bandwidth to transmit matrix M: The value of Dt(M) can be computed using size of matrix M, and size of each element in M. $Dt(M) = El \times m \times n$, where El is data size of each element in matrix and m, n are size of matrix M, respectively. For example, when El=2 bytes, The value of each element range from -32768 to 32767.

Piconet (P_i) : A piconet is represented as $P_i = \{(M_i, S_{i,j}) | 1 \le j \le 7\}$, where M_i denotes master of piconet P_i and $S_{i,j}$ 則 denotes the j-th slave in the i-th piconet.

2.2 Strassen Algorithm

Strassen Algorithm was proposed in 1977 to perform efficient matrix multiplication. Strassen algorithm decomposes matrix multiplication $A_{m,p}$ and $B_{p,n}$ into 7 sub-matrix multiplication problem where the size of matrix is half of original matrix. Each sub-matrix multiplication is calculated individually. The results of these seven sub-matrix multiplication are then integrated to form the product matrix $MAB_{m,n}$. The details and time complexity analysis of Strassen algorithm are described as following. Finally, we will use divide and conquer approach to extent Strassen algorithm.

A. Operations of Strassen Algorithm

For the sake of description, symbols applied in Strassen algorithm are summarized as follows.

QA_{mn}: Block partitions of matrix A. Strassen algorithm partitions a large matrix into four blocks of matrices with equal sizes. Assume the size of original matrix A is $n \times n$, where n=2^k. Block partitions of matrices are represented as $\begin{bmatrix} QA_{11} & QA_{12} \\ QA_{21} & QA_{22} \end{bmatrix}$, where QA_{11}, QA_{12}, QA_{21} and QA_{22} are block partitions of matrices with sizes of $\frac{n}{2} \times \frac{n}{2}$.

B. Time Complexity Analysis of Strassen Algorithm

Let $n = 2^k$, where k is an integer. For matrix multiplication problems with size $n \times n$, pre-additions step need ten matrix addition/subtraction with size $\frac{n}{2} \times \frac{n}{2}$. Recursive calls step require 7 matrix multiplications with size $\frac{n}{2} \times \frac{n}{2}$. Post-additions step require 8 matrix additions/subtractions. Generally speaking, Strassen algorithm needs 7 $\frac{n}{2} \times \frac{n}{2}$ matrix multiplications, and 18 $\frac{n}{2} \times \frac{n}{2}$ matrix additions/subtractions. The time complexity of Strassen algorithm is O($n^{\log 7}$).

3.Wireless Parallel Matrix Multiplication Algorithm (WPMM)

Conventional parallel processing operates in a wired communication environment. For parallel processing in wireless environment, simultaneous communications among wireless devices is an important research issue. The proposed wireless parallel matrix multiplication algorithm operates in a divide and conquer manner. The algorithm comprise scattering, computation and gathering phases. WPMM algorithm adopted Strassen algorithm to divide a large matrix multiplication problem into 7 matrix-multiplications sub-problems with half size of original one.

3.1 Extension of Strassen Algorithm

The operations of Strassen algorithm are summarized in Table 1. From Table 1, it is observed that the large matrix multiplication problem $A \times B$ is decomposed into seven sub-matrix multiplication problems (the computation of matrix QP_1 to QP_7. To obtain the product of $A \times B$, further matrix additions/subtractions are required. Using divide and conquer mechanism, the computation of QP_i can be recursively decomposed into smaller matrix-multiplication sub-problems. This process can be repeated until a suitable size of matrix is obtained to perform matrix multiplications.

Table 1: Summary of operations in Strassen algorithm.

$QP_1 = (QA_{11} + QA_{22}) \times (QB_{11} + QB_{22})$	
$QP_2 = (QA_{21} + QA_{22}) \times QB_{11}$	
$QP_3 = QA_{11} \times (QB_{12} - QB_{22})$	$QC_{11} = QP_1 + QP_4 - QP_5 + QP_7$
$QP_4 = QA_{22} \times (QB_{21} - QB_{11})$	$QC_{21} = QP_2 + QP_4$
$QP_5 = (QA_{11} - QA_{12}) \times QB_{22}$	$QC_{12} = QP_3 + QP_5$
$QP_6 = (QA_{21} - QA_{11}) \times (QB_{11} + QB_{12})$	$QC_{22} = QP_1 - QP_2 + QP_3 + QP_6$
$QP_7 = (QA_{12} - QA_{22}) \times (QB_{21} + QB_{22})$	

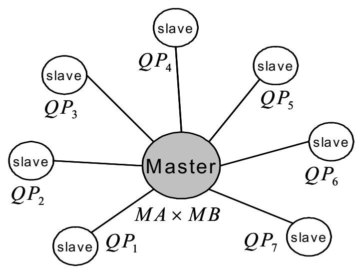


Figure 1: The relation between matrix multiplication and Bluetooth piconet.

3.2The relation between Strassen algorithm and Bluetooth Piconet

The divide and conquer mechanism is applied to compute matrix multiplication. Let $C_m = A_m \times B_m$, where $m = 2^k$, $k \in N$ and $m \geq 2$. From Table 1, it is observed that to obtain the product C_m, the sub-matrix QP_1 to QP_7 must first be computed. In Bluetooth, the problem can be described as follows. A master needs to compute matrix multiplication $A_m \times B_m$, it can decompose this problem into 7 sub-problems QP_1 to QP_7 and distributes these sub-problems into its seven slaves. Figure 1 shows the relation between a matrix multiplication problem and a Bluetooth piconet. When a slave receive sub-matrix, it can decide further decomposition is possible or not. If it is possible to further decompose a sub-matrix into smaller sub-matrix, it can recursively decompose the problem and distribute these sub-matrix into its seven slave. If further decomposition is impossible, it will perform matrix multiplication in this device directly. When result in each slave is computed, the results should send to master device to compute the final matrix *C*. Total processing steps can be categorized into *scattering*, *computing*, and *gathering* phases. In scattering phase, master divide large matrix multiplication problem into seven matrix multiplication sub-problems. Master then send needed sub-matrix to each slave. When the size of matrix is equal to predefined threshold, the slave will turn to computing phase. It is assumed that the computing power of each device is the same, so the computing time in computing phases in each device is the same. After computing phase, each device will turn to gathering phase. In gathering phase, slave can obtain results through inter-slave communications, so that the computing time can be shorten.

4. Bluetooth-based Wireless Parallel Matrix Multiplication Algorithm (BWPMM)

To reduce interference, Bluetooth use frequency-hopping spread spectrum (FHSS) mechanism. Each device will hop within 79 channels. It is possible that more than one piconet exists in the same place. So, multiple device pair within different piconets can communicate simultaneously. As shown in Figure 2, BWPMM algorithm includes scattering, computing, and gathering phases. First, a large matrix multiplication is divided into 7 matrix multiplication sub-problems. The division process repeats until a suitable size of matrix is reached. Each device then compute results of matrix multiplication. Finally, the results are integrated to obtain solution of original matrix multiplication problem.

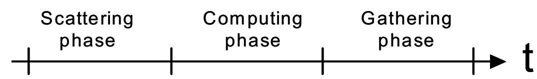


Figure 2: Three phases of Bluetooth-based Wireless Parallel Matrix Multiplication (BWPMM) algorithm.

4.1 Detail operations of BWPMM

A. Scattering Phase

CP(P_i): Probability of collision within a Piconet。

TS_i: Average time of transmitting matrix in Scattering Phase. TS_i can be computed as follows.

$$TS_i = Dt(QP) + Dt(QA \text{ or } QB) \times CP(P_i)$$

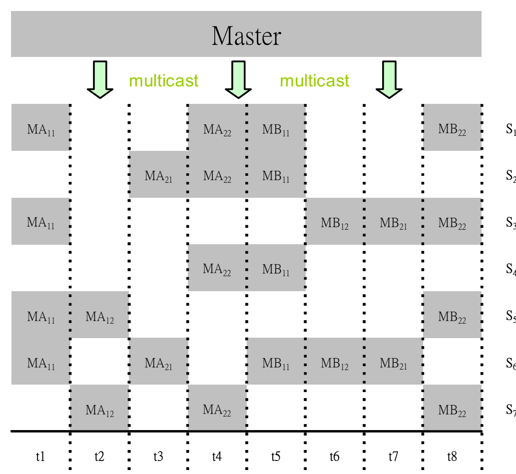


Figure 3: Operations in Scattering Phase.

The objective of Scattering phase is to divide matrix into smaller sub-matrix and send to each slave. From Table 1, it is observed that the computation of QP_1, QP_3, QP_5, and QP_6 all need QA_{11} as operand. To reduce bandwidth consumption, master in piconet can use multicast communicaiton to transmit sub-matrix to all salves. Figure 3 demonstrates operations of scattering phase. Initially, master determines the size of TS_1 to TS_8 according to size of matrix and available communication bandwidth. Time slots TS_1, TS_2, ... , TS_8 are used to transmit QA_{11}, QA_{12}, …, QB_{22}, respectively. Master can determine the size of each time interval and broadcast this time schedule to all slaves. A piconet contains at most seven slaves. There is a naturally association between a slave S_i and a sub-matrix multiplication QP_i, as shown in Table 1 and Figure 2. For example, slave S_1 is responsible to compute $QP_1 = (QA_{11} + QA_{22}) \times (QB_{11} + QB_{22})$, So S_1 needs to receive sub-matrices, QA_{11}, QA_{22}, QB_{11}, and QB_{22}. The sub-matrices where each slave needs to receive are denoted by gray area as shown in Figure 3. Master broadcasts each sub-matrix sequentially. Each slave can either receive sub-matrix or enter power saving state. In Scattering phase, Time slots TS_1 to TS_8 are used to transmit sub-matrices four sub-matrices in A and four sub-matrices of B. At the end of TS_8, extra time slots can be reserved for data retransmission if packet lost were occurred. During transmission, packet type DH5 is used to increase transmission efficiency.

B. Computing Phase

If a slave is an M/S relay, it will further decompose the received sub-matrix QPi into smaller sub-matrices and perform role switching operations to become master of another piconet. It will repeat scattering, computing and gathering phase again. However, if role of a device is pure slave, the device will perform matrix multiplication directly.

C. Gathering Phase

The ultimate goal of BWPMM is to compute the final product matrix MAB. When results QP_i in Slave S_i are computed, Master device needs to integrate QP_1 to QP_7 to obtain QC_{11}, QC_{12}, QC_{21}, and QC_{22}. If multi-channel characteristics of Bluetooth can be used to speedup communications, the computation time of QC_{11}, QC_{12}, QC_{21}, and QC_{22} can be reduced significantly. In the following, a greedy algorithm is proposed to reduce the number of sub-matrix transmission between master and slave from 7 to 5. The number of matrix additions/subtractions can be reduced to 4.

4.2 Greedy Algorithm

Greedy algorithm use the following rules to choose communication pairs among slaves to reduce communication time.

Rule 1: Immediate Rule

Because master device can communicate with one slave at a time, the utilization of master device should as high as possible, so that performance within a piconet can be increased. When one slave has computed results of QC_{11}, QC_{12}, QC_{21}, or QC_{22}, it should transmit this result to master as soon as possible. Master thus can obtain QC_{11}, QC_{12}, QC_{21}, and QC_{22}.

Rule 2: Prepare Rule

To increase utilization of master device and avoid unnecessary waiting due to master is busy, the prepare rule is defined. In prepare rule, some slaves are chosen to compute MC_{11}, MC_{12}, MC_{21}, or MC_{22} as early as possible, so that the communication between master and slave can be performed as soon as possible.

Rule 3: Broadcast Rule

In Bluetooth, data exchanging among slaves can be achieved efficiently by packet broadcasting. So, in gathering phase, some QP matrix transmission can be performed through broadcasting, so that the number of matrix transmission can be reduced from two to one.

Rule 4: Reduce Rule

The computation of QC_{11} requires three matrix transmission, i.e. $QC_{11}=QP_1+QP_4-QP_5+QP_7$. If QP matrix can be transmitted as soon as possible, the waiting time of Master can be reduced to compute QC_{11}.

The priorities of the above rules are as follows: rule1 > rule2 > rule3 > rule4. Figure 4 illustrates operations of the proposed greedy algorithm. The time needed to transmit QP matrix in gathering phase is defined as Tg_i.

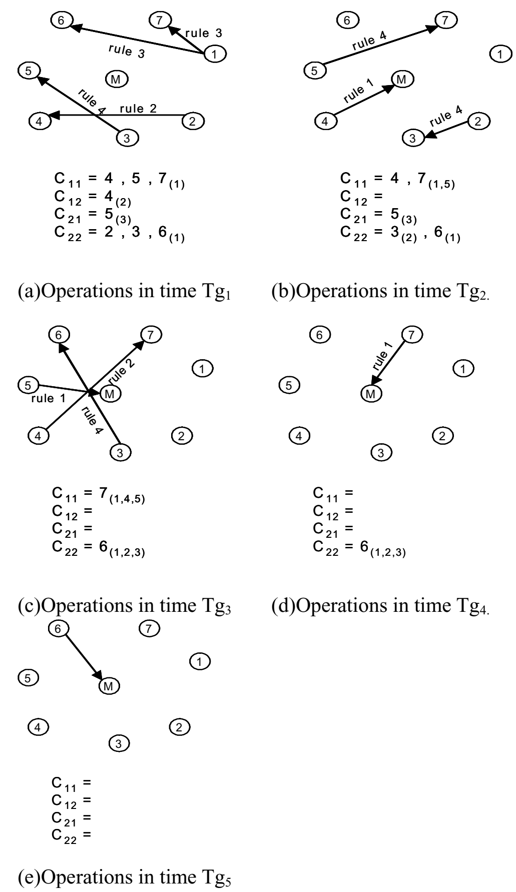


Figure 4: Operations of the Greedy algorithm.

4.3 Parallel Communication

In gathering phase, there are lots of parallel transmissions need to be done. For example, in Figure 4(a), Slave 2 transmits QP_2 to Slave 4, Slave 1 transmits to Slave 6, and Slave 3 transmits to Slave 5, simultaneously. These parallel transmissions can be achieved through role switching operations. First, Slave 1, 2, and 3 send role switching LMP request packets to Master. Then, master replies accept packets to these slaves. Information contained in LMP acceptance packet includes BD addresses and clock offsets of Slave 4, 5, 6, and 7. After Slave 1, 2, and 3 received LMP accept packets, they can enter page state using previous collected BD addresses and clock offsets. Master inform slave 4, 5, 6, and 7 to enter page scan state. A scatternet is then constructed as shown in Figure 5. In Figure 5, the gray nodes indicate masters of temporary constructed piconets.

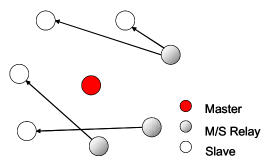


Figure 5: Illustration of Piconet restructuring.

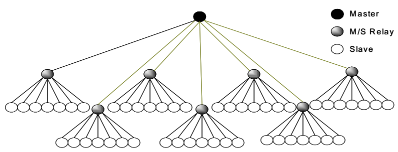


Figure 7: A scatternet is connected as a hierarchical tree structure with degree 7.

4.4.Perform Strassen Algorithm on BT Tree

Due to the constraint that a piconet has at most seven active slaves, the proposed Bluetooth-based parallel matrix multiplication algorithm adopts Strassen style computation paradigm. As shown in Figure 6, a Bluetooth scatternet contains a tree structure with branches 7 in each internal node. A large matrix multiplication problem in parent node is decomposed into seven small matrix multiplication problems computed in seven slave nodes.

5. Experimental Results

To evaluate performance of BWPMM algorithm, Bluetooth devices involved in parallel computation are organized as a hierarchical tree structure. This tree can be used to observe performance of divide and conquer parallel algorithm. Initially, all devices stay in inquiry scan states. The device which initiate parallel computation task will change its state to inquiry state. When the first piconet with one master and seven salves is constructed, these slaves will change state to inquiry state and construct piconets with seven slaves in the second level of BT tree. These process repeats until the designated level is constructed or all devices are connected. In the following, all experiments use Strassen algorithm to compute matrix multiplication and use Bluetooh-N protocol with no collision to evaluate system performance. Related parameters used in experiments are summarized as follows.

1. The size of each element in a matrix is 2 bytes.
2. The bandwidth of wireless communication is assumed to be 79 MB/s.

The computing powers of most mobile devices are weak. In the following experiments, it is assumed that the computing power of mobile device is 100MIPS, so that the relation between the number of devices and the required computing time can be established. Theoretically, increasing the number of devices will decrease the required computing time. However, when the number of devices increases from 64 to 512, the computing time does not decrease. The reason why such situation happens relies on the fact that the level of tree increases and the dimension of matrix in leaf nodes decreases when the number of devices increase. The computing tasks of BWPMM are executed in leaf nodes. When the number of devices increase, the reduction of computing time in leaf nodes can not compensate the increasing of communication overhead.

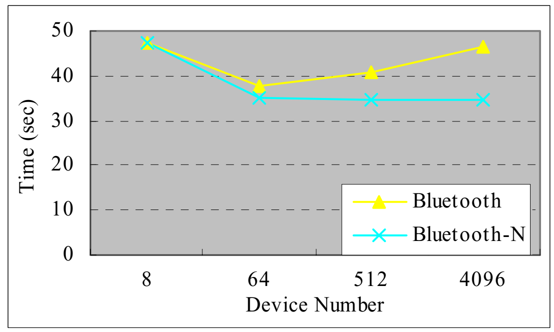


Figure 7: The relation between the number of computing devices and total computing time when the bandwidth is 79MB.

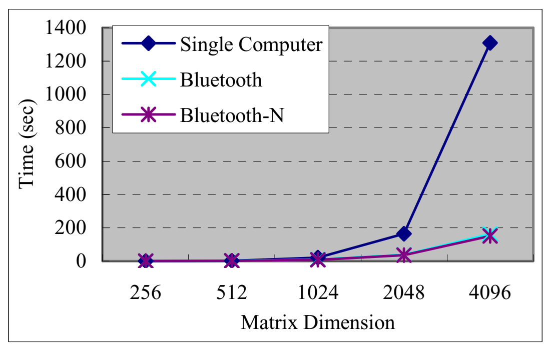


Figure 8: The relation between the dimension of matrix and total computing time.

As the dimension increased, the communication time of a matrix transmission also increased. As shown in Figure 8, the total computing time of the task increases when the size of matrix is increased. The reason why the cost increases is that the times to perform computation and communication of matrix multiplication both are increased. As the computing power of a mobile device increases, the communication capability remains the same. As shown in Figure 9, when the computing power of a device is increasing k times, the needed computing time decreases to $1/k$ times. For parallel computation task, the computing power of mobile device has small impacts on the total computing time. In this circumstance of low computing power, the number of devices used in the computing process has more inferences on the computing and communication times.

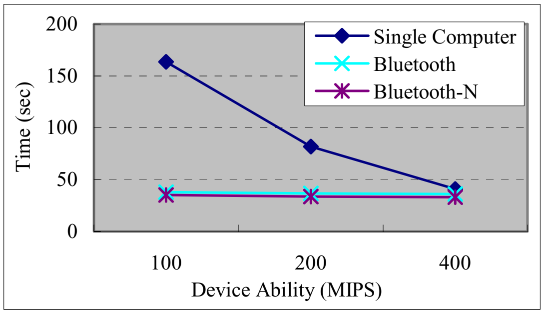


Figure 9: The relation between computing power of device and total computing time.

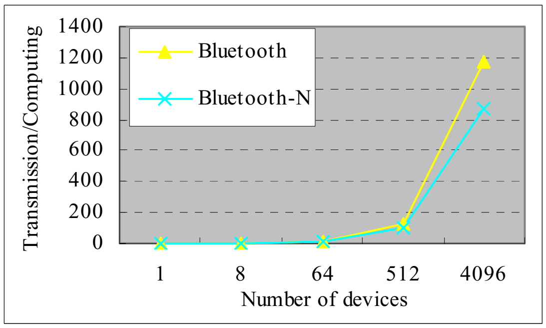


Figure 10: The relation between the number of devices and the ratio of communication over computing time.

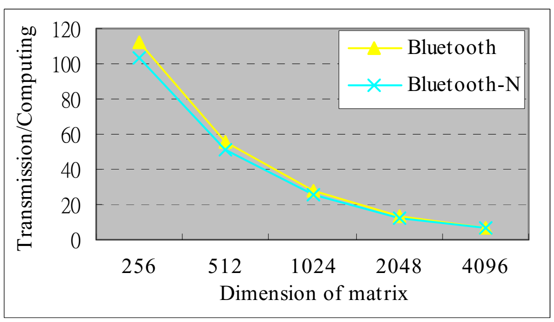


Figure 11: The relation between the dimension of matrix and the ratio of communication over computing time.

As shown in Figure 10, when the number of computing devices is increased, the ratio of communication over computing time will also increased even if the computing power of mobile device remains the same. The reason why this situation occurs is that increasing the number of computing devices will reduce the size of matrix multiplications computed in leaf nodes and increase the height of hierarchical tree architecture simultaneously. When the size of matrix multiplication is reduced, the computing time overhead is reduced. When the height of hierarchical tree structure is increased, the communication overhead is increased.

From the simulation results shown in Figure 11, when the dimension of matrix is increased, the ratio of transmission over computing time is reduced during the parallel computing process. This fact reveals that increasing the size of matrix will also increase the computing time. It is conjectured that when the size of matrix is increased, the increasing computing time is twice as the increasing communication time. Because large matrix multiplication problem will reduce the impacts of communication overhead, large matrix multiplications will benefit the BWPMM algorithm.

6. Conclusion and Future Work

In this paper, we have proposed a Bluetooth-based wireless and parallel matrix multiplication (BWPMM) algorithm. The BWPMM algorithm use divide-and-conquer mechanism and hierarchical tree network architecture to reduce computing time of large matrix multiplication problem. As demonstrated by experimental results, the proposed BWPMM algorithm indeed reduces the computing time and increases bandwidth utilization. The future work of this paper will try to discover more efficient wireless transmission technologies to improve system performance and apply the proposed.

Acknowledgement

The author would like to thank the National Science Council of the Republic of China for financially supporting this research under Contract No. NSC93-2524-S-156-001 and NSC 93-2213-E-156-001.

References

[1]. C. Y. Chang, K. P. Shih, C. H. Tseng, C. F. Wang,"A Role Switching Agent for Reducing Packet Lost Phenomenon in Bluetooth Wireless Networks, " in *Proceedings of the 8th International Conference on Distributed Multimedia Systems (DMS'2002)*, pp. 473-479, San Francisco, USA, Sep. 2002.

[2]. Chih-Fang Wang, Sahni, S., "Matrix multiplication on the OTIS-Mesh optoelectronic computer," in *IEEE Transactions on Computers*, Volume: 50 , Issue: 7 , July 2001 , Pages:635 - 646

[3]. Lih-Chyau Wu, Shing-Tsan Huang, A new parallel computation model on synchronous wireless interconnection , *IEEE TENCON '93 Conference on Computer, Communication, Control and Power Engineering*, Region 10, Issue: 0 , 19-21 Oct. 1993

[4]. Chao-Tsun Chang, Chih-Yung Chang, and Jang-Ping Sheu, "BlueCube: Constructing a Hypercube Parallel Computing and Communication Environment over Bluetooth Radio System," in *Proceedings of the 2003 International Conference on Parallel Processing*, October 06 - 09, 2003, Kaohsiung, Taiwan.

A Power-Efficient Multi-Sink Routing Protocol for Wireless Sensor Networks

En-Yi Chien and Gwo-Jong Yu
Department of Computer and Information Science,
Aletheia University, Tamsui, Taipei, Taiwan
yugj@email.au.edu.tw

Abstract

The major power consumption of wireless sensor node comes from data collection process from sensor node to sink node. Because the total energy of a mobile device is limited, the life time of a mobile device lasts until all of its power is exhausted. In this paper, an energy efficient anycast routing protocol is proposed. The proposed protocol finds a routing path with minimum accumulated power consumption to transmit a data packet from sensor node to sink node. When a sensor node moves, leaves, joins, or enters low power state, new routing paths can be maintained through local flooding process. As demonstrated by experimental results, the proposed anycast routing protocol indeed increase the lifetime of entire network.

Keywords – Anycast routing protocol, network lifetime, wireless sensor network.

I. INTRODUCTION

A sensor network consists of sensor and sink nodes. Operator can handle and manage sensor network through sink nodes. Operations of sensor network include event detection, temporary data storing, and data forwarding. When a sensor node is deployed, the lifetime of this node lasts from power on to power exhausted. Due to the energy of sensor node is limited, the communication range of a sensor node is small in most applications. Many power consumption researches [3][4][5] assume that power consumption is direct proportional to square of communication distance. Besides, a sensor node can not move by itself unless external force is put on this device. If some sensor node is moved by external force, the detection range and accuracy of data collection will be poor. To insure that the events within a range can be completely collected, the number of sensor nodes should be large enough.

In literature, routing algorithms can be categorized into the following classes [1]: (1) Sensor node chooses the routing paths with minimum hop count. Small number of devices involved in this class of routing algorithm. The message can be sent to sink node with small delay time. However, to reduce the number of hop count, each sensor node may choose a device with the longest distance within a hop. In this circumstance, the power of sensor node is exhausted fast. When some device exhausts its power, a device may not find a routing path to sink node even if most of sensor nodes are alive in a sensor network. (2) Sensor node can forward message to sink node through a routing path based on principle of minimum energy consumption. The drawback of this class of methods is that when two sensor nodes are very close, routing path pass through nearby devices will always choose these two devices. So the power of these two devices will exhaust fast. (3) Each sensor node computes remaining power. Devices with high remaining power have large probabilities to be chosen to forward message. The advantage of this method is that the remaining powers of most of sensor node are power balanced. The drawbacks of this method are as follows. Some devices with large remaining power will involve in the routing path even if it is not necessary to this node to forward messages or a longer routing path is chosen to forward message. These drawbacks will exhaust extra power. All these three class of methods have advantages and drawbacks. Which class of routing algorithm should be chosen to perform routing path establishment depends on various conditions. To take new sensor node into considerations, the second class of routing algorithms is selected in this paper.

The remaining sections of this paper are organized as follows. In section II, we will describe the relation between power consumption and communication distance and details parts of the proposed protocol. Performance analysis is conducted in Section III to verify the proposed protocol. The protocol proposed in [8] will be used as benchmark to evaluate performance of the proposed protocol. Finally, we will conclude this paper in Section IV.

II. ENERGY-EFFICIENT ANYCAST ROUTING PROTOCOL

In this section, the relation between power consumption and communication distance of sensor nodes will be described first. Then the data structure and message type will be summarized to describe the proposed protocol. Section 2.1 describes the detail operations of the proposed protocol. In Section 2.2, route maintenances tasks for predictable or unpredictable topology change of networks will be described. Scenario, problems, and solutions of various route maintenance tasks are presented in details to illustrate how routing path is maintained. The power consumption of message transmission between two sensor nodes can be formulated as:

$$E_T(k, d) = E_{elec} * k + E_{amp} * k * d^2 \qquad (1)$$

$$E_R(k) = E_{elec} * k \qquad (2)$$

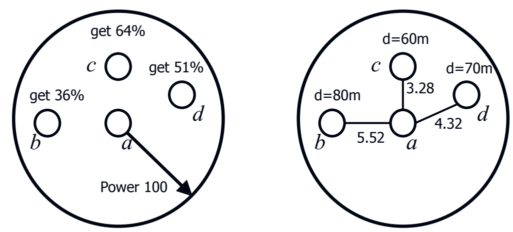
In Equation (1) and (2), E_T and E_R represent power consumptions for transmitting and receiving a message for a sensor node. Either transmitting or receiving messages need to consume power E_{elec}. In transmitting message, E_{amp} represents power consumption constant for amplifier circuit. Typical values of E_{elec} and E_{amp} are E_{elec}=50*nJ/bit*, E_{amp}=100*pJ/bit/m*^{2}. Parameters k and d represents message length and communication distance, respectively.

Based on Equations (1) and (2), the distance between two sensor nodes can be estimated from transmission and receiving power levels. The internal state of a sensor node may be working or idle. In working state, a sensor node is active and this node can transmit and receive messages. Working state can be further partitioned into awake and low power states. When the power level of a sensor node is within 10%~100% of full power, a sensor node is in awakening state. In awaken state, a sensor node can perform all operations. When the power level of a device is below 10% of full power, it will stay at low power state. In low power state, a node can only perform sensing, transmitting, and receiving operations. Hello and info_request messages will be ignored. Idle state is used to reduce power consumption. It is not necessary for most wireless sensor network applications to keep sensor node awake all the times. In this circumstance, a sensor node can enter power saving mode to reduce power consumption. In idle state, a sensor node can monitor medium through overhearing mechanism. If the content is relevant to a sensor node, it can switch from idle to awake mode to processing the following tasks.

When a sensor node is deployed, some information is maintained for future routing path decision and inter-node communication. Each node maintains the following data structure. (1) ID: Each sensor node owns a unique ID which is used to identify itself when this node communicates with other nodes. (2) hop_count: Sink node broadcasts hello message periodically. Each node will extract hop count from message, increase the hop count and rebroadcast the updated message. Based on hop count information, a sensor node knows how to route a message to sink node. (3) total_cost: This field contains the total power consumption when a message is send from this node to sink node. This power consumption is computed by this node and all relay nodes from this node to sink node. (4) hop_cost: Hop cost is the power consumption to transmit message from this node the next hop node. (5) next_node: Next node is ID of the neighboring device which is used to relay message from this node to sink node. (6) energy: energy field indicates the remaining power level of this sensor node. (7) cost: Assume there are m nodes within communication range. The fields *cost*[*i*][1] are used to record ID of the *i*-th neighboring nodes. The fields *cost*[i][1] are used to record the power consumption when a message is transmit from this node the *i*-th neighboring sensor node.

2.1 Route Establishment

In sensor network, when several sensor nodes are deployed, it is necessary to connect these nodes to form a topology, so that neighboring events can be detected and transmitted to sink node. Figure 1 shows an example.



(a) When node *a* transmits message, the strength of message is dominated by communication distance.

(b) neighbor nodes of *a* compute the power consumption between node *a* and itself.

Figure 1: Topology construction through interconnected sensor nodes.

Initially, each sensor node operates independently and connects with no sensor node. When a sensor node intends to establish connection with other nodes, it will broadcast a hello message with fixed power level *P*. According to the received power level, neighboring nodes can estimate distance between sender and receiver nodes. Neighboring nodes can compute the required power consumption when a message is send from sender to this node and vise versa. All sensor nodes will broadcast hello messages, so a sensor node knows how many nodes in its neighboring and how much power consumption when a neighboring nodes send message to itself. In this way, a connected network topology can be constructed. To find routing path from sensor node to sink node, an on-demand like routing protocol is proposed in this paper. The following routing information is maintained in each node: (1) ID of next hop nodes in routing path. (2) Accumulated power consumption when this routing path is chosen to delivery message.

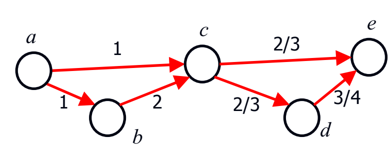


Figure 2: Illustration of unnecessary power consumption.

When sink nodes broadcast message, the broadcast storm problem [9] should be taken into considerations. Another condition which consumes extra power is demonstrated in Figure 2. When *a* sends a message, *b* and *c* will receive the message simultaneously. In next turn, *b* sends message to *c* and *c* send message to *d* and *e*. In this circumstance, if device *d* and *e* find that forwarding message through *b* is a better choice, they will update their table again. To reduce the effect of this problem, we propose to defer message rebroadcasts to reduce power consumption.

2.2 Route Maintenance

When sensor nodes are deployed in outside environment, it is possible that some external force will change network topology. When this happens, it is necessary to update routing paths for changing nodes to make the entire network operates normally. In the following, we take four scenarios into considerations and propose solutions to solve these problems.

2.2.1 Mobility Management

When a sensor node is moved by some external force, it is said to be a moved sensor node and abbreviated as MSN. When a MSN is moved, one of the following conditions may happen. (1) The distance between MSN and nearest next hop node is shorten and the distance between MSN and previous hop is increased. In this circumstance, message from previous hop can not delivery to this hop and message from this node to next hop is delivered with larger power consumption. (2) MSN can not send message to next hop using the power level computed by sink flooding. All routing paths passing through this MSN need to find new routing paths.

If all nodes are mobile, i.e. in a mobile sensor network, the routing path can be fixed by increasing the power level, Max(last cost, next cost), of MSN to transmit message. Using a larger power level, MSN can delivery messages to both previous and next hop nodes. The message received by next hop node can be forwarded to sink node. The message received by previous hop can be treated as Acknowledge message of previous message transmission to ensure a reliable message transmission. If sensor node have transmitted a message and have not received acknowledge message in time out interval, this node need to send info_request message to neighbor nodes. Neighboring nodes will response message to this node to update routing path information.

2.2.2 Sensor Leave

If a sensor node is destroyed by external force and lost its operation capabilities, this node can be treated as leaving sensor network. The leaving node is said to be a leaved sensor node (LSN). When a sensor node is leaving, all routing path passing through this LSN become invalid. For example, if *S*3 needs *S*4 to relay message to sink node and *S*4 becomes a LSN, then *S*4 will not forward messages sent by *S*3. In this circumstance, *S*3 sends message to *S*4 and receive no acknowledge message from *S*3. *S*3 thinks that *S*4 is either moved or leaved. *S*3 will broadcast info_request to find new routing path to sink nodes. After received response from neighboring nodes, *S*3 can find new routing path to sink node.

2.2.3 Sensor Join

As the sensor network operates for a long time, some sensor nodes will exhaust its power. When a lot of sensor nodes have no power, the sensor network may become partitioned and can not operate normally. To solve this problem, redeployment of new sensor nodes can make the network continuously operation. When new sensor nodes join sensor network, two conditions may occur. In the first condition, only small number of new sensor nodes join sensor network. The new nodes is said to be a joined sensor node (JSN). When JSN joins the network, it will first broadcast a hello message. The neighboring nodes compute power consumption when transmit message between itself and the JSN, includes its information into hello_reply message, and reply hello_reply message to JSN. JSN can use information in hello_reply message to find best routing path. From this example, we can observe that new joined JSN can mitigate power consumption and improve lifetime of entire network.

Another condition is that sensor nodes consume their powers in similar manner. In this circumstance, many sensor nodes may exhaust their power simultaneously. To keep the network continuously operation, many JSNs should be added to the sensor network simultaneously. When these new JSNs join sensor network, they may broadcasts info_requests to get information of neighboring nodes. But the neighboring nodes are JSN too, and will not reply hello_reply to info_request message, so JSN can not obtain routing path information.

2.2.4 Sensor Low Power

The fourth condition is sensor low power. This condition is similar to LSN, but the difference is that LSN is unpredictable and sensor low power is predictable. When the power level of a sensor node is low, it can compute the required sharing of power consumption by neighboring nodes to increase lifetime of sensor node without adding new sensor nodes. When the power level of a sensor node is below a predefined threshold L, it is said to be a low power sensor node (LPSN). To reduce power consumption, LPSN will execute sensing and transmission tasks and will not relay message of other sensor nodes.

III. Simulation

To evaluate performance of the proposed multi-sink routing protocol, a simulation is implemented using C++ language. In simulation, the positions of sink and sensor nodes are randomly selected. To obtain meaningful results, simulations are performed several times with various randomly generated topologies. Average of several results is computed as the final result. The protocol proposed in [8] is also implemented as benchmark and is denoted as ARMN. For the sake of reasonable comparison, only the condition of leaving sensor nodes is taken into consideration. When a routing path is found, a sensor node is randomly selected to generate message and this node will forward message along the computed routing path. The message is transmitted 2000 times and results are averaged to computed mean values. The simulation parameters are summarized as in Table 1.

Table 1: Parameters used in simulation

Number of sensor nodes	20
Number of sink nodes	3
Number of message transmission when a routing path is constructed.	2000
Transmission range	100m
Simulation area	200m*200m
Initial power	200J

Figure 3 shows the relation between operation time and number of remaining active nodes. In the same environment, after 2000 message transmissions, the proposed protocol has 6-14 active nodes which is better than ARMN which has 4-12 active nodes.

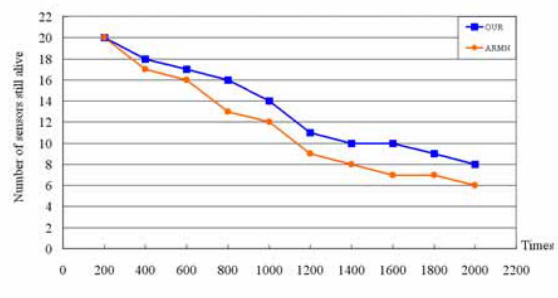


Figure 3: The relation between operation times and number of remaining active nodes.

Figure 4 shows accumulated hop counts after 2000 message transmission. From Figure 4, it is observed that the proposed protocol require 300 more hops than ARMN.

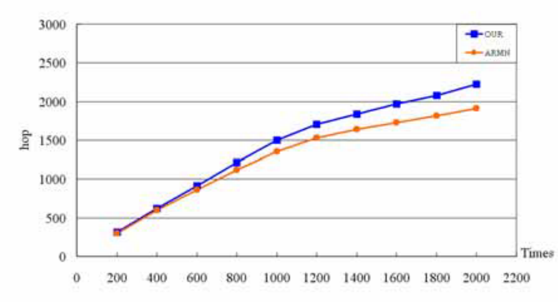


Figure 4: Accumulated hop count after 2000 message transmissions.

Figure 5 demonstrates the relations between operation time and total remaining energe. The proposed protocol remains 200J more power than ARMN after 2000 message transmissions.

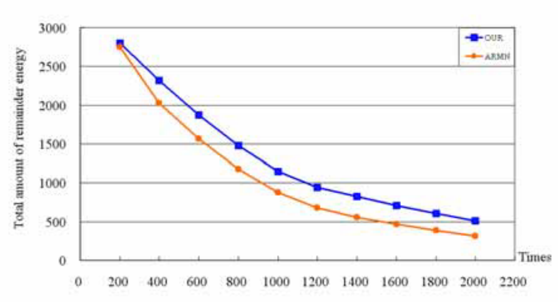


Figure 5: The relation between operation time and remaining power of entire network.

IV. Conclusion

In this paper, we have proposed a multi-sink routing protocol which can find routing path with minimum accumulated power consumption. Solutions of scenario such as moving, join, leaving, and low power of sensor nodes are also proposed to enhance the network lifetime. From simulation results, the proposed protocol has better performance than ARMN in terms of active remaining nodes and accumulated remaining powers. The proposed protocol indeed can enhance lifetime of sensor network. In future work, the proposed protocol will be extended to cluster-based network architecture to further reduce power consumption.

Acknowledgement

This work was partially supported by learning technology, sponsored by the National Science Council, NSC 93-2213-E-156-001 and NSC 93-2524-S-156-001

References

[1] Akyildiz, I.F., Weilian Su, Sankarasubramaniam, Y., amd Cayirci, E., "A survey on sensor networks", *Communications Magazine, IEEE*, Volume: 40, Issue: 8, Pages: 102 – 114, Aug. 2002

[2] Tubaishat, M., and Madria, S., "Sensor networks: an overview", *Potentials, IEEE*, Volume: 22, Issue: 2, Pages: 20 – 23, April-May 2003

[3] Heinzelman, W.R., Chandrakasan, A., and Balakrishnan, H., "Energy-efficient communication protocol for wireless microsensor networks", System Sciences, 2000. *Proceedings of the 33rd Annual Hawaii International Conference on*, Pages: 10 pp. vol.2, 4-7 Jan. 2000

[4] "Power Efficient Data Gathering and Aggregation in Wireless Sensor Networks", In *Special section on sensor network technology and sensor data management*, Volume 32, Issue 4, Pages: 66 – 71, December 2003 SPECIAL ISSUE: Special section on sensor network technology and sensor data management

[5] Jain-Shing Liu, and Lin, C.-H.P., "Power-efficiency clustering method with power-limit constraint for sensor networks", *Conference Proceedings of the 2003 IEEE International*, Pages:129 – 136, 9-11 April 2003

[6] Raju, J., and Garcia-Luna-Aceves, J.J., "A comparison of on-demand and table driven routing for ad-hoc wireless networks", *2000 IEEE International Conference on*, Volume: 3, 18-22, Pages:1702 - 1706 vol.3, June 2000

[7] Lee, S.-J., Gerla, M., and Toh, C.-K., "A simulation study of table-driven and on-demand routing protocols for mobile ad hoc networks", *Network, IEEE*, Volume: 13, Issue: 4, Pages:48 – 54, July-Aug. 1999

[8] Park, V.D., and Macker, J.P., "Anycast routing for mobile networking", Military Communications Conference Proceedings, 1999. *MILCOM 1999. IEEE*, Volume: 1, 31, Pages:1 - 5 vol.1, Oct.-3 Nov. 1999

[9] Min-Te Sun, and Ten-Hwang Lai ,"Location aided broadcast in wireless ad hoc network systems", *Wireless Communications and Networking Conference, WCNC2002*. 2002, IEEE. 17-21 March 2002 Pages:597 - 602

A Location-aware Multicasting Protocol for Bluetooth Location Networks

Chih-Yung Chang, Kuei-Ping Shih, Wen-Chuan Huang, Shih-Chieh Lee, Hsu-Ruey Chang, Kuang-Jen Liao
Department of Computer Science and Information Engineering,
Tamkang University, Tamsui, Taipei, Taiwan
Email: cychang@mail.tku.edu.tw

ABSTRACT

Bluetooth Location Network (BLN) [2] is a mobile network that consists of a number of location-aware Bluetooth devices. In a BLN, a multicast service is provided by sending messages from a service server to a number of specific members via a Bluetooth wireless network. This paper proposes a multicasting protocol for constructing efficient multicast tree in BLN. Adopted with role switching [1][3] and time-slot leasing[4][5] techniques, the constructed multicast tree reduces the transmission delay and power consumptions for multicast services. Additionally, a maintenance protocol is proposed to resolve the route disconnection problem causing by node mobility reason, preventing the constructed multicast tree from broken. The proposed protocol will construct a multicast tree with good features of shortest route, high degree of path sharing, and fewer forwarding nodes. Simulation results reveal that the proposed multicast tree outperforms the multiple unicast routing and traditional multicast protocols in the BLN.

Keywords: Bluetooth, Bluetooth Location Networks, multicast, role switch, time leasing.

1. INTRODUCTION

Bluetooth [1] is a promising technology that aims at supporting wireless connectivity among various mobile devices such as cellular phones, headsets, PDAs, and their peripheries. Device equipped with Bluetooth chips are able to communicate with other devices with characteristics of low power, low costs, short distances, in a wireless manner.

The smallest network unit in Bluetooth system is a piconet, which consists of one master and ups to seven active slaves. By connecting multiple piconets, a large-area network can be formed called scatternet. In a scatternet, source and destination may exist in different piconets. An efficient routing protocol is required to support cross piconet data communication service. A number of routing protocols have been proposed to construct a route from source to destination. Most of them adopt a flooding scheme where a source broadcasts a route search packet and node in network rebroadcast the received route search packet in advance. On receiving the route search packet, the destination node replies with a route reply packet to construct a route. Although the flooding operation may generate a shortest route, it consumes scatternet bandwidth and devices' power. Some other researches developed location methods to provide each device with the location information. The location information helps not only to find the location of event but also reduce the overhead raised by flooding.

Multicast service has been widely discussed in both wired and wireless networks. The trend of installing wireless equipment such as Bluetooth in mobile devices Cellular phone, PDA, and other portable devices will increase the need of providing mobile users with multicast service. These requirements will especially be found in a large area public service region such as mall, shop, airport, and so on. The construction of Bluetooth Location Network (BLN) has been discussed in [2], which is an architecture system of environment designed for obtaining the indoor location information of mobile devices. By the cooperative computing of installed Bluetooth access points, BLN provides location information of Bluetooth mobile devices, thus supporting user the services such as obtaining instant information, peer-to-peer communication, or query information by using the equipped Bluetooth mobile device. Network resources thus can be highly utilized.

This paper aims at developing efficient multicast protocol for providing multicast service in a BLN. With the assumption that the BLN provide each mobile device the location information, the paper proposes an efficient multicast protocol with good features of shortest route, high degree of path sharing, and least forwarding nodes.

The remainder of this paper is organized as follows. Section 2 introduces the Bluetooth and Bluetooth Location Networks system and states the problems investigated in this article. The terminology used in this paper is provided in Section 3. Section 4 describes in details the proposed relative coordinates-based multicasting protocol (RCMP) and a maintenance protocol. Performance simulations of the proposed RCMP are discusses in Section 5 and Section 6 finally concludes the paper.

2. BACKGROUND AND RELATED WORKS

2.1 Bluetooth

Bluetooth operates in the international 2.4 GHz Industrial, Scientific and Medical (ISM) band, at a gross data rate of 1Mbit/second, and features low energy consumption for use in battery operated devices. To achieve the highest possible robustness for noisy radio environments, Bluetooth uses a packet-switching protocol based on a frequency hop scheme with 1600 hops per second. The entire available frequency spectrum is used with 79 hops of 1 MHz bandwidth. Virtual channels are defined using pseudo-random hop sequences.

In the Bluetooth network all units are peer units with identical hardware and software interfaces distinguished by a unique 48-bit Bluetooth device address (BD_ADDR). At the start of a connection, the initializing unit is temporarily assigned as a master and the other one is assigned as a slave. This assignment is valid only during this connection. Slaves are assigned a temporary 3-bit active member address (AM_ADDR) by master to reduce the number of addressing bits required for active communication. A piconet is the smallest network unit formed by a master and one or up to seven active slaves in Bluetooth. Each piconet is defined by a different frequency hopping channel which is the pseudo-random hopping sequence of the piconet master. All units participating in the same piconet are synchronized to this channel sequence. If a node participated in more than one piconet and responsible for relaying data from one piconet to another, it is named bridge (or relay). Two different types of bridge are existed in scatternet: m/s bridge, plays a master role in any participated piconet, and s/s bridge.

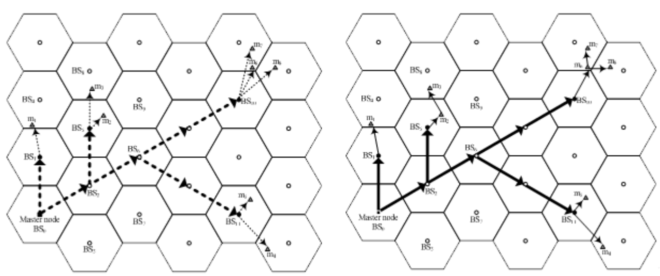
2.2 Bluetooth Location Networks

González-Castaño and García-Reinoso [2] proposed a Bluetooth Location Network (BLN) for location-aware or context-driven mobile networks, such as m-commerce networks of e-museums. In such scenarios, there exist a service server (or several service servers) that needs to know user location in real-time, to send context-oriented information to user handhelds when necessary. The sensing location information is transmitted automatically in BLN without user participation.

The BLN is composed by mobile badges and static Bluetooth units, which establish a spontaneous network topology at system initialization. BLN users carry either Bluetooth-enabled handheld or any mobile data terminal with a Bluetooth badge. Static Bluetooth units, we will refer to the latter in this thesis as *Bluetooth Stations* (BSs), are small, wireless Bluetooth nodes, which are arranged in a network that covers the whole target area.

2.3 Problem statement

The BLN architecture is proposed to apply in various scenarios, such as mall, shop, and so on, where the service server needs to transmit some data packets to several mobile users. Thus an efficient multicast routing protocol is necessary. However there are no multicast routing protocol has been proposed in the BLN. The multicast routing protocols [6][7] designed for 802.11-based wireless networks are widely investigated in the past years but all of them is unsuitably implemented in Bluetooth.



(a) Communicated multicasting routes (b) Physical multicasting routes

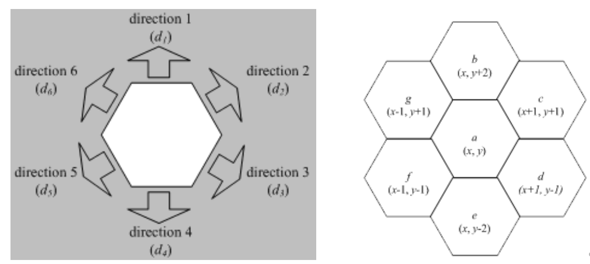
Figure 1: The results of an example for executing the relative coordinates-based multicasting protocol

The paper proposed herein is developed based on Bluetooth Location Network environmental system. A multicast service is considered in BLN. There is a service server and some particular mobile multicast members. The service server is the source node which has a sequence of messages to transmit to all members whereas the multicast members are mobile and intend to receive the multicast message from service server. Each mobile device will be detected by Bluetooth Station and the location information will be transmitted to the service server using the shortest path. Therefore, the service server has each member's location, which significantly decreases the control overheads from discovering and establishing route among service server and all members. The proposed two-stage multicasting protocol uses distributed on-demand manner to establish the multicast forwarding routes without using excess control packets. The constructed multicast routes, as show in Fig. 1, has good features of the shortest route between each source-destination pair, high degree of path sharing, less forwarding nodes, and low control overhead for route maintenance.

3. TERMINOLOGY AND TABLES

In this section we define the terminology and tables used in the remainder of the paper to facilitate the explanation.

- **Physical and communicated multicasting routes**
  The constructed multicasting route is distinguished into two routes, physical route and communicated route. The physical route stands for the physical link existed between two devices and communicated route stands for the packets transmitted route. The finally communicated multicasting routes of an example are showed in Fig. 1(a) and Fig. 1(b) shows the finally physical multicasting route.



(a) Directions (b) Relative coordinates

Figure 2: The directions and relative coordinates of each cell

- **Responsible Member (*rm*)**
  Each member receives data packets generated by service server from a specific BS. The member *i* receives data packets from BS *a* is called the responsible member of BS *a* and named $rm_i(BS_a)$.

- **Home Bluetooth Station (*H-BS*)**
  The BS, which is indicated to connect with members and responsible for forwarding packets by the service server, is called *H-BS*.

- **Candidate Home Bluetooth Station (*CH-BS*)**
  In the BLN environment, a mobile Bluetooth device may be detected by several BSs. One or more of these BSs which has the shortest hop to service server is referred to as *CH-BS*.

  Notice that *S*(*BS*) stands for the set of BS, and the relation of *S*(*BS*), *S*(C*H-BS*), and *S*(*H-BS*) is $S(BS) \supseteq S(CH-BS) \supseteq S(H-BS)$. In other words, if a BS is the H-BS, it is also the CH-BS. But if a BS is the CH-BS, it is not represent that the BS is the H-BS.

- **Direction (*d*)**
  Assumption each cell has six directions, d_n where $1 \leqq n \leqq 6$. As Fig. 2(a) shows, d_1 stands for the north direction and the remaining *n* value is increased clockwise.

- **Relative Coordinates (*RC*(*x*, *y*))**
  The coordinates of each cell equals the coordinates of each BS because of each cell just has included one BS. The coordinates of the BS_a (or the cell *a*) and the relative coordinate to BS_a of BS_b is represented as $C_{BS_a}(x, y)$ and $RC_{BS_a}^{BS_b}(x, y)$ respectively.

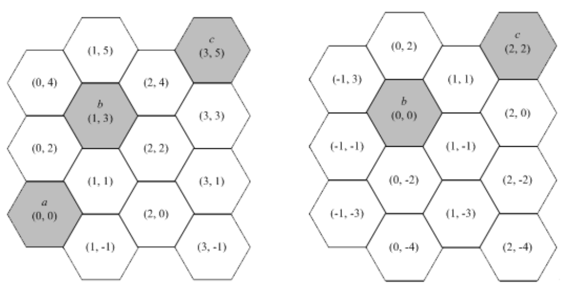
Assumption the coordinates of cell a is $C_{cell_a}(x,y)$, as show in Fig. 2(b), the relative coordinate to it of neighboring cells are $RC_{cell_a}^{cell_b}(x,y+2)$, $RC_{cell_a}^{cell_c}(x+1,y+1)$, $RC_{cell_a}^{cell_d}(x+1,y-1)$, $RC_{cell_a}^{cell_e}(x,y-2)$, $RC_{cell_a}^{cell_f}(x-1,y-2)$, and $RC_{cell_a}^{cell_g}(x-1,y+1)$ corresponding to d_1 to d_6, respectively.

For example, in Fig. 3(a), the relative coordinates to cell a of cell b and cell c is $RC_{cell_a}^{cell_b}(1,3)$ and $RC_{cell_a}^{cell_c}(3,5)$, respectively. We can obtain the following equation from observing these cells' relative coordinates:

$$RC_{cell_i}^{cell_j}(x_1,y_1) - RC_{cell_i}^{cell_k}(x_2,y_2) = RC_{cell_k}^{cell_j}(x_3,y_3)$$, where $x_3 = x_1 - x_2$, $y_3 = y_1 - y_2$.

According the above equation, we can obtain the relative coordinates to cell b of cell c is $RC_{cell_b}^{cell_c}(2,2) = RC_{cell_a}^{cell_c}(3,5) - RC_{cell_a}^{cell_b}(1,3)$, which conforms to the fact show in Fig. 3(b).

- **Member Table**
  The member table records the served multicast group members of each H-BS which is created in the initialization of multicast service. Each H-BS and its responsible members are recorded in corresponding fields. The relative coordinates to current BS of each H-BS is used to calculate the H-BS's located direction.



(a) Relative coordinates to cell a of cell b and cell c

(b) Relative coordinates to cell b of cell c

Figure 3: Relative coordinates

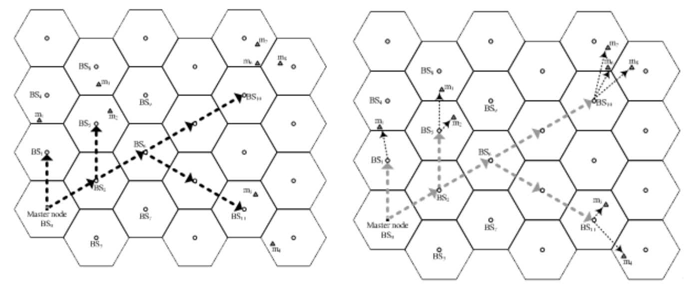
- **Direction Table**
  The direction table is created in each forwarding BS to decide the packet forwarding direction. The direction of each ABS is estimated through proposed BSLD algorithm, detail in next section. The next hop field records the BS corresponded to direction field.

- **Multicast Routing Table**
  A multicast routing table is created on demand and maintained by each BS. The multicast group id and sequence number are used to identify the multicast group and detect duplicates, respectively. The forwarded next BS is record in next hop and the backup direction field records the backup route. TTL is used to determine the entry deletion time.

4. RELATED COORDINATES-BASED MULTICASTING PROTOCOL

This section introduces the proposed Related Coordinates-based Multicasting Protocol (RCMP). The DMRP consists of two phases, the *multicast routes construction* (*MRC*) *phase* and the *role assignment* (*RA*) *phase*. The former phase mainly constructs the trunk of multicast routes among BSs, whereas the later phase establishes the links among BS and mobile members.

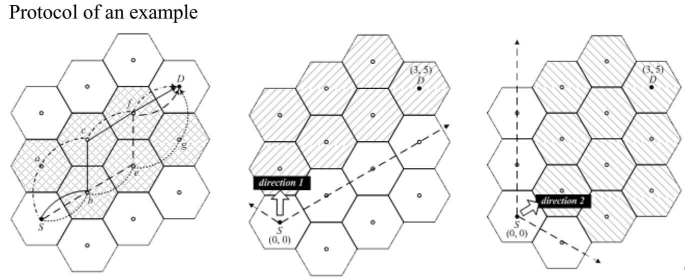
In the MRC phase, six directions and comparative coordinates of each cell is adopted to construct the communicated routes between Master node and each H-BS. Using distributed manner calculates the optional packet forwarding directions and establishes the shortest routes between computing BS and H-BS. In other words, this phase establishes the trunk of the multicast routes with features of shortest hops, high degree of sharing path, and least forwarding nodes. Fig. 4(a) illustrates the resulted multicast tree after executing the MRC phase of an example.



(a) The executing results of the MRC phase

(b) The executing results of the RA phase

Figure 4: The executing results of the Relative Coordinates-based Multicasting Protocol of an example



(a) The shortest routes between S and D

(b) The target area of direction 1 for S

(c) The target area of direction 2 for S

Figure 5: The concepts of the Candidate Direction Calculation algorithm

The RA phase is used to establish the link between H-BS and its responsible members. According to the specification of Bluetooth radio system, each master just can connect with seven active slaves. In the BLN, however, each BS has connected with six active slaves, the six neighboring BSs. If a BS played master role to connect with any group member, it will lose the capability of sensing function. Therefore, using *role switching* (*RS*) operation [1][3] to exchange the roles of H-BS and its responsible member is proposed in RA phase to solve this problem. Furthermore, the *time-slot leasing* (*TSL*) [4][5] technology is adapted to save delay time and power consumption. Fig. 4(b) illustrates the connected multicasting links established in the RA phase of an example. The finally physical multicasting routes and connected multicasting routes are shown in Fig. 1.

4.1 Candidate Direction Calculation Algorithm

Before formally details the RCMP, a proposed algorithm used to calculate the optimal packet forwarding direction for each H-BS, named *Candidate Direction Calculation* (*CDC*) algorithm, is introduced first. The "optimal direction" stands for "the route established in this direction with features of shortest distance between current node and destination node." Fig. 5 illustrates the concepts of this algorithm.

We use an example to describe the calculation of CDC algorithm subsequently. For example, in Fig. 5(b), the relative coordinates to cell *S* of cell *D* is (3, 5). According to the calculation rules of CDC algorithm, y=5≧2 and y=5=(x+2×1)∈{5, 7, 9, 11, …}, where x=3, cell *D* located in the target area of direction 1 for cell *S*. Cell *D* also located in the target area of direction 2 for cell *S* because of x=3≧1 and y=5=((-x)+2×4)∈{-1, 1, 3, 5, …}. Therefore, the candidate forwarding for cell *S* to transmit packets to cell *D* are direction 1 and direction 2. The calculated results of CDC algorithm correspond with the facts shows in Fig. 5(a) and (b).

The proposed CDC algorithm will calculate the located direction of each H-BS and indicate that which direction is the indication of shortest routes between current BS and H-BS. This algorithm is adopted in MRC phase of the RCMP, detailed in next subsection, used to calculate the candidate forwarding direction in order to construct the shortest routes among service server and each H-BS.

4.2 Multicast Routes Construction Phase

This subsection details the first phase, *multicast routes construction* (*MRC*) phase, of the proposed RCMP. The MRC phase is mainly aim to construct the multicast routes between Master node and each H-BS. Therefore, the real locations of all members are ignored temporarily in this phase and the route between each H-BS and its responsible members will be established in the second phase.

In BLN environments, a member may be detected by several BSs and the detected information will be replied to Master node. Therefore, all members' information is collected in Master node and the location information will be calculated via simple compute by service server. When service server wants to start the multicast service and transmit amount of data packets for members, it executes the following step to initiate the RCMP:

Step 1: ***H-BS selection and member table creation*:** The H-BS is indicated to connect with members and is responsible for forwarding multicast packets to them. In the initiation of MRC phase, service server chooses the CH-BS and H-BS for each member. For each member *i*, the CH-BS is the detecting BS. If there exists more then one BS is its CH-BS, a CH-BS which detected the maximal number of members is the H-BS; otherwise, the CH-BS is the H-BS of member *i*. In other words, member *i* is the responsible member of its H-BS. Each H-BS and its responsible members are recorded in member table.

When BS receives the multicast packet first time, includes Master node, it executes the following steps to establish the multicast route. We use the following notations to facilitate the description of these steps.

- BS_i = the BS which implements the MRC phase.
- D = a set of six directions.
- $S_{H\text{-}BS}(d_n)$ = a set of H-BSs where the direction d_n is the candidate forwarding direction of these H-BSs.

Step 1. ***Relative coordinates update*:** In accordance with transmitted direction of received packet, the BS_i updates the relative coordinates of each $H\text{-}BS_j$ and replace the original *Relative coordinates* field in the received member table. The Master node is unnecessary to implement this step because it is the source of the multicast packet.

Step 2. ***Candidate forwarding direction calculation and direction table creation*:** The BS_i takes the relative coordinates of each $H\text{-}BS_j$, $RC_{BS_i}^{H-BS_j}(x, y)$, recorded in received member table into CDC algorithm and then gets candidate forwarding direction(s) form the output. Recording each $H\text{-}BS_j$ and its candidate forwarding direction(s) in the corresponding fields of direction table.

Step 3. ***Multicast routing table creation*:** The BS_i executes the following steps to create the multicast routing table.

(1) Select a direction d_n, where $d_n \in D$ and $|S_{H\text{-}BS}(d_n)|$ = max.

(2) Record d_n and corresponding H-BSs into multicast routing table. If $H\text{-}BS_j$ has two candidate forwarding directions, where $H\text{-}BS_j \in S_{H\text{-}BS}(d_n)$, records another direction in backup direction field; otherwise, the backup direction field sets null.

(3) Fill other fields of the multicast routing table with corresponding information indicated in the multicast packet, such as multicast group id, sequence number and TTL.

(4) Remove d_n and corresponding H-BSs from direction table.

(5)Repeat above steps until the set D is empty.

Step 4. ***Data packet transmission*:** After multicast routing table creation, the BS_i transmits the multicast packet, which includes member table, to its neighboring downstream node according to multicast routing table.

The *H-BS selection* step in the service server is used to reduce the number of unnecessary H-BS. The *direction table creation* step adopts the CDC algorithm which is based on the concepts of relative coordinates to calculate the optimal packet forwarding direction(s) for each H-BS. The multicasting route established in this direction(s) has the future of shortest hop between current BS and H-BS. To centralize the routes between Master node and each H-BS is implemented in the *multicast routing table creation* step. Therefore, the MRC phase implemented by each forwarding BS establishes the multicasting routes with features of shortest hops, high degree of sharing path, and least forwarding nodes.

Notice that the multicast routes construction process is implemented when service server transmits or the BS receives the first multicast packet. For subsequent multicast packets, each forwarding BS just forward them to the downstream BSs according to the multicast routing table and it is unnecessary to implement above steps expect reseating the TTL value.

4.3 Role Assignment Phase

The MRC phase constructs the multicast routes from Master node to each H-BS. In *role assignment* (*RA*) phase, the link between

H-BS and member is established and it is the part of the multicasting routes. The *role switching* operation [1][3] and *time-slot leasing* technology [4][5] are adopted to solve the internal limitations existed in the BLN environment and to reduce the transmission delay and power consumptions for multicast services.

In the BLN environment, each BS not only plays the master role to control six surrounding slave BSs, but also plays the slave role which is controlled by the neighboring six master BSs. The Bluetooth standard specifies that the maximum of active slaves which connected with a master node is seven, and a slave node can be connected with unlimited number of master nodes. Therefore, the BS only remains one available connection which is used to sense the surrounding Bluetooth devices (member). There are existing following limitations in this situation.

***Limitation 1*: The trade-off between sensing function and member connection**

If the BS decides to establish the connection with the member, it would loss the function of sense, and contrariwise. A simple solution to solve this trade-off limitation is using park mode. Let six surrounding slave BSs into park mode in turn to keep there is always existing one available connection for the BS, which plays the master role. In this situation, the BS can establish the connection with member without losing the sensing function. But the rise in delay of the sensing information report and packet transmission will occur subsequently.

***Limitation 2*: Connected members switch**

If the BS decided to establish the connection with the member and the number of member is more than one, it would incur the second limitation. There is only one available connection can be established with the member but several members need to receive packets from the same BS. The simple solution is also using the concept of park mode. The BS establishes the connection with all members and keeps that there is only one connected slave in active mode. However there exist the same disadvantages as the limitation 1.

In RA phase, the main object is to establish the connection between each H-BS and its responsible members. When the H-BS receives the multicast packets with new multicast group id, it implements the RA phase to complete the multicast tree construction. We use the following notations to facilitate the description of these steps.

- $H\text{-}BS_i$ = the BS which implements the RA phase.
- $S_{rm}(H\text{-}BS_i)$ = the set of members which are the responsible members of $H\text{-}BS_i$.

Step 1. ***Main connection selection***: The $H\text{-}BS_i$ use page process to establish the link with a responsible member rm_i. In this moment, $H\text{-}BS_i$ plays the master role and rm_j plays the slave role. It there are no other responsible members exists, then goes to Step 3. Otherwise the $H\text{-}BS_i$ executes the next step.

Step 2. ***Member-to-member connection request***: The $H\text{-}BS_i$ sends the $mmpeg\_req(S_{rm}(H\text{-}BS_i)\text{-}\{rm_j\})$ packet to responsible member rm_j for requesting it try to establish the connection with other unconnectedly responsible members. The packet includes the BD_ADDRs and clocks of all responsible members except rm_j.

Step 3. ***Role switching execution***: The $H\text{-}BS_i$ executes role switching process to form a new piconet p_i, where rm_j is the master and $H\text{-}BS_i$ becomes the slave in p_i.

Step 4. ***Member-to-member connection***: The responsible member rm_j attempts to establish the connection with each responsible member, where $rm_j \in S_{rm}(H\text{-}BS_i)$, when it receives the $mmpeg\_req(S_{rm}(H\text{-}BS_i)\text{-}\{rm_j\})$ packet from the $H\text{-}BS_i$. Once the rm_j successfully establish the connection with a responsible member rm_k, it removes rm_k from $S_{rm}(H\text{-}BS_i)$ set. After a round of attempt, rm_j replies a $mmpeg\_rep(S_{rm}(H\text{-}BS_i))$ packet to the $H\text{-}BS_i$ for reporting the member-to-member connection results.

Step 5. ***Time-slot leasing execution***: The $H\text{-}BS_i$ initiates the TSL process to lease time slots from rm_j. In the duration of leasing slots, the $H\text{-}BS_i$ is the temp-master and other connectedly responsible members are the temp-slaves of p_i.

Step 6. ***RA phase completion checking***: The $H\text{-}BS_i$ checks the connection results with the received $mmpeg\_rep(S_{rm}(H\text{-}BS_i))$ packet. If there still has unconnected responsible member, the $H\text{-}BS_i$ goes bake to Step 1. Otherwise the $H\text{-}BS_i$ completes the RA phase.

The RS process executed in Step 3 is used to solve the limitation 1. The H-BS changes the role form master to slave, therefore the active connection umber of its piconet remains to six. The sensing function is held and the packet transmission delay is not increase. Step 5 employs the TSL technology to solve the limitation 2. In the slot-leasing duration, the H-BS leases slots from responsible member to transmit packets to several responsible members at the same time. TSL technology was, moreover, employed to change the authority of polling but doesn't change the original roles. Therefore, the H-BS still plays the slave role but has the authority to transmit multicast packets without waiting responsible member's polling. The data transmission delay will decrease outstandingly.

4.4 Maintenance Protocol

The implementation of the RCMP constructs a multicast route with good features of shortest route, high degree of path sharing, and less forwarding nodes. Members may move everywhere inside the BLN, therefore, a maintenance protocol to keep up the connection between service server and member is necessary. The maintenance protocol introduced in this subsection is proposed to handle the route disconnection problem causing by member mobility reason, preventing the constructed multicast tree from broken.

The maintenance protocol can be divided into two viewpoints, member and BS. In member's viewpoint, when member discovers that the signal strength from H-BS is below than a predefined threshold value, it sends an *information request* packet to the H-BS for acquiring neighboring BSs' information, includes BD_ADDR, clock, and signal strength. The signal strength between member and BS is obtained during sensing process. The member establishes the connection with a BS with the highest signal strength, named *New H-BS*, and prepares for cutting the connection with the H-BS. Once the member receives the new multicast construction information from the New H-BS, it will disconnect the original multicast route, and the New H-BS is responsible to forward the multicast packets to the member.

In the BS viewpoint, when H-BS discovers the signal strength form responsible member is below than a threshold value or receives the *information request* packet form the member, it will send the information of neighboring BSs to the member and prepare for removing the connection with it. If a BS connects with a new member, it will become a H-BS and send the *route reconstruction* packet along the route to the service server. When a BS received the *route reconstruction* packet, it checks that it is the part of multicast routes or not. If the BS is, it will reply a *reconstruction ack* packet to establish the additional multicast route and report this information to the service server. Otherwise, the BS will forward the reconstruction packet to its upstream node along the route to the service server.

5. PERFORMANCE STUDY

To evaluate the performance of proposed RCMP, we have implemented it based on the BlueHoc simulator [8], which is an ns2 [9]-based simulator released by IBM. Our simulation modeled a Bluetooth Location Network of 114 prearranged Bluetooth Stations and 30 mobile multicast members in a 100x100 meter square. Radio propagation range of each Bluetooth node is 10 meters, and nominal channel capacity is set to 1Mbps. All members are randomly generated and roam randomly in all directions at a predefined average speed from 0 to 9 m/s. The simulation time is 100 seconds. The traffic source of service server is constant bit rate source (CBR) sending with a rate of 10 data packets per second. The data packet size is 2870 bits, DH5 packet, and DH1 packet, 366 bits, is used for control packet transmission.

In the simulation experiments, four transmission manners are compared, Flooding, Multiple Unicast Routing Protocol, Traditional Multicast Routing Protocol, and proposed Relative Coordinates-based Multicasting Protocol. Average delay and throughput are two comparing metrics.

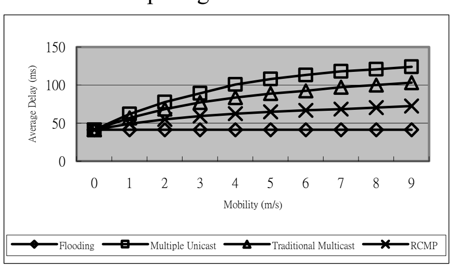


Figure 6: Average delay comparison over mobility

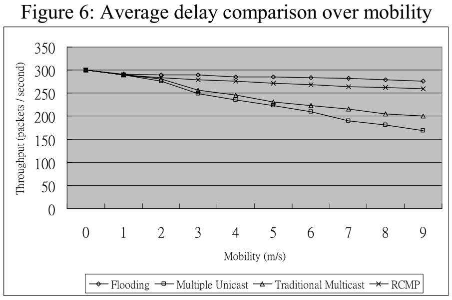


Figure 7: Throughput comparison over mobility

Fig. 6 shows the average delay of comparing schemes over mobility. As the mobility increases, the average delays of all schemes are increased excluding the Flooding scheme. All schemes necessarily repair the disconnection routes causing by mobility reason, and the Flooding scheme is unnecessarily. The Multiple Unicast scheme needs longer time period to reconnect the routing path than the Traditional Multicast scheme; therefore, the average delay of it is higher. Similarly, the average delay of the Traditional Multicast scheme is higher than the RCMP because of the longer time of path repair.

Fig. 7 compares the throughput of Flooding, Multiple Unicast, Traditional Multicast and RCMP. We note that for zero mobility (static network) all schemes achieve the same throughput. As mobility increases, the throughputs of all schemes are decreased. The Multiple Unicast and Traditional Multicast schemes yields lower throughput because the effects of route reconstruction. Both of them need a space of time to repair the disconnect route, especially the Multiple Unicast scheme. Flooding scheme always keeps the highest throughput because it unnecessarily creates specific routes to all members. There are various routes existed between Master node and each member can be used to transmit data packets by the Flooding scheme.

6. CONCLUSIONS

This paper proposed a Candidate Direction Calculation (CDC) Algorithm and a Relative Coordinates-based Multicasting Protocol (RCMP) over Bluetooth Location Networks. The CDC algorithm using the concepts of relative coordinates to calculate each H-BS's located direction and indicate the direction of shortest path for packets forwarding. The RCMP constructs a multicast tree with good features of the shortest route between each source-destination pair, high degree of path sharing, less forwarding nodes, and low control overhead for route maintenance. Simulation results show that the proposed RCMP scheme outperforms than the multiple unicast routing and distributed multicast routing in all comparing metrics. In average delay, and throughput, flooding has the highest performance because the route construction of it is unnecessary. But it needs higher multicast service cost than the proposed RCMP. If there exist more than one service serve in the BLN, using flooding manner may occur the network congestion problem and reduce the network life time.

REFERENCES

[1] Bluetooth Specification Version 1.1, http://www.bluetooth.org. Feb. 22, 2001.

[2] F. J. González-Castaño and J. Garcia-Reinoso, "Bluetooth Location Networks," *IEEE Global Telecommunications Conference* (*GLOBECOM'02*), pp. 223-237, Nov. 2002.

[3] C. Y. Chang, K. P. Shih, C. H. Tseng, and C. F. Wang, "A Role Switching Agent for Reducing Packet Lost Phenomenon in Bluetooth Wireless Networks," *The 8th International Conference on Distributed Multimedia System* (*DMS'02*), pp. 473-479, Sep. 2002.

[4] Wensheng Zhang, Hao Zhu, and Guohong Cao, "Improving Bluetooth Network Performance Through a Time-Slot Leasing Approach," *IEEE Wireless Communication and Networking Conference* (*WCNC'02*), pp. 592-596, Mar. 2002.

[5] C. Cordeiro, S. Abhyankar, and D. P. Agrawal, "A Dynamic Slot Assignment Scheme for Slave-to-Slave and Multicast-like Communication in Bluetooth Personal Area Networks," *IEEE Global Telecommunications Conference* (*GLOBECOM'03*), pp. 4127-4132, Dec. 2003.

[6] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," *IEEE Workshop on Mobile Computing System and Applications* (*WMCSA'09*), pp. 90-100, Feb. 1999.

[7] Elizabeth M. Royer, Charles E. Perkins, "Multicast Operation of the Ad-hoc On-Demand Distance Vector Routing Protocol," *ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 207-218, Aug. 1999.

[8] Kumar, A., BlueHoc Manual, http://oos.software.ibm.com/bluehoc/tutorial/docpage.html. IBM India Research Lab.

[9] http://www.isi.edu/nsnam/ns. NS. The network simulator.

GUARD: A GUide, Alarm, Recovery, and Detection System on a Wireless Sensor Network for the Blind

Kuei-Ping Shih, Chih-Yung Chang, Chien-Min Chou, Shih-Chieh Lee,
Yen-Da Chen, and Hsu-Ruey Chang

Department of Computer Science and Information Engineering
Tamkang University, Tamshui 251, Taipei, Taiwan
{kpshih,cychang}@mail.tku.edu.tw

Abstract

This paper proposes a GUARD (GUide, Alarm, Recovery, and Detection) system for the blind. The main goal of GUARD system is to develop a wireless sensor network in order to guarantee the safety and the convenience for the blind in Tamkang University. We will implement the hardware, the firmware, the deployment algorithm, the route discovery/rediscovery algorithm, the positioning algorithm, and the application programs of sensor nodes which will be applied to the GUARD system for the blind. The GUARD system aims at providing guide, alarm, recovery and detection capabilities in Tamkang University to construct an obstacle-free environment for the blind. For the sake of the GUARD system, we design and implement several communication protocols and programs in MICA2 Motes in order to design the GUARD system. We also implement the guiding system that allows the blind to be tracked and guided in the campus of Tamkang University.

I. Introduction

Hardware and wireless technologies have great advanced in recent years. Sensor nodes therefore with low cost, small volume, multi-functionalities, low power can be easily implemented. Wireless Sensor Networks (WSN) which is composed of many wireless sensor nodes can be widely used in varied applications, such as disaster reporting, military detection and monitoring, location tracking, environment sampling, healthy monitoring, natural ecology monitoring, and so on. The system for the blind in Tamkang University had been constructed for years. Much experience is gained with computers for the blinds, the construction of guide bricks, and the use of guide dogs. As the development of technology, the system provides more secure, timely, and varied services for the blind. This paper tries to use the previous experience to conduct the wireless and sensor technology in order to provide a GUide, Alarm, Recovery, and Detection System (GUARD) on a Wireless Sensor Network for the Blind. This paper takes Tamkang University as the basic scenario to construct a WSN for the blind. Since GUARD is used for the blind, energy consumption should be taken into consideration when constructing a WSN. Such a WSN can serve for the blind as long as possible. Therefore this paper tries to construct a WSN with longer lifetime. Besides, GUARD wishes to provide accurate position information for the blind. When the blind is walking in the campus, they can know their locations at any given time to increase their sense of security. Consequently, GUARD will initiate the WSN in order to know the position information of each sensor nodes.

In the infrastructure-less WSN, the location of sensor nodes are computed in advance before constructing a WSN. Sensor nodes then build the network topology by themselves. As the differences of the sensing applications, every WSN should take different metrics into account. Therefore, Andrew Howard et al. [1] assumed that sensor nodes have the moving capability. They tried to maximize the coverage area by the self-moving of sensor nodes which can move to the un-coverage area and extend the sensing field. Li Li et al. [2] proved that every node transmits with the minimum power required to ensure that in every cone of degree = A is some nodes that can reach. $A = 5/6\pi$ can guarantee that network connectivity is preserved. Consequently, [2] not only ensures the network connectivity, but also decreases the power consumption in order to extend the lifetime of a WSN. F. Mondinelli et al. [3] used sink to localize all sensor nodes in a WSN. After the topology of a WSN is constructed, the best paths for each sensor nodes to disseminate data can be found. Errol L. Lloyd et al. [4] tried to compute the transmission power of every sensor nodes in order to minimize the total energy consumption in a WAN. Himanshu Gupta et al. [5] proposed a centralized algorithm to solve the problem of sensing some region in a WSN. Through the centralized algorithm, the sensing region will find a best subset of sensor nodes to perform the sensing request. Other redundant sensor nodes do not need to sense. Thus this algorithm can reduce the energy consumption.

Besides, there are many researches focus on irregular hierarchical WSNs. This kind of researches try to construct clusters in WSNs. How to select the appropriate cluster heads, how to increase the lifetime of this kind WSNs, and where to put a sink are the main challenges of this kind of researches. Gaurav Gupta et al. [6] proposed a algorithm to control the topology of a WSN. Because sensor nodes who are elected as cluster heads will consume their energy rapidly. [6] tries to balance the number of cluster members between every clusters. Consequently, every cluster head will have closer life time after balancing their cluster members. However, [6] do consider how to elect cluster heads. Wendi Rabiner Heizelman et al. [7] proposed an algorithm to solve this problem. Every sensor node in [7] has chance to be a cluster head. Hence the energy consumption of sensor nodes will become equally.

The rest of this paper is organized as follows. Section II is the overview of the GUARD system. In Section III, we describe the deployment method of the GUARD system. Section IV is the rewriting of sensor nodes. Section V concludes this paper.

II. Overview

We use Macromedia® FLASH™[8] as the GUI (Graphic User Interface) in the GUARD system. The main considerations are as follows:

- FLASH is a vector based drafting system. Consequently, the map in FLASH system will not lose reality when we are enlarging or shrinking it.
- FLASH performs well in multimedia. We can easily develop effects and visual messages in dynamic map.
- FLASH can integrate with the database system completely. Users can authenticate through FLASH and can manipulate the GUARD system through network in real-time manner.

Fig. 1 shows the construction of the GUARD system in Tamkang University. We constructed a WSN in Tamkang University. The GUI of the GUARD system is shown in Fig. 2. When the blind walks in the campus. The blind can use a walking stick with a embedded sensor node. Through the communication between sensor nodes, users (such like security guards, system administrators) therefore

Fig. 1. Constructing a WSN in Tamkang University.

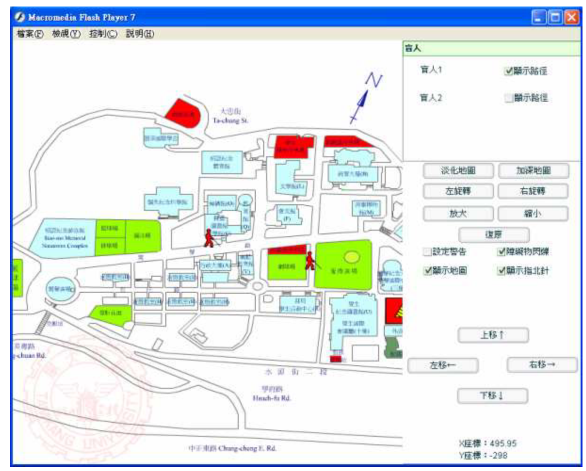


Fig. 2. GUI of the GUARD system.

can know the position of the blind. Fig. 2 can be divided into two parts: the control panel part and the map part. In the control panel part, users can configure what they want. The result of configuration will show in the map part according to the setting of users. There configurations include blind position, route planning, obstacles information, and etc. 3 shows the protocol stack of the GUARD system. Detailed functionalities of the GUARD system are shown as follows:

- When a blind is walking in the campus, he can set his destination where he wants to go to. When the WSN in Tamkang is initializing, position of every sensor node can be known either through the capability of localization in LLC sub-layer or through robots when deploying. The GUARD system can compute the position of a blind through the positioning algorithm in Network layer. Therefore the GUARD system can calculate the shortest path for the blind and **guide** the blind to his destination.
- When a object (a car or something else) is moving

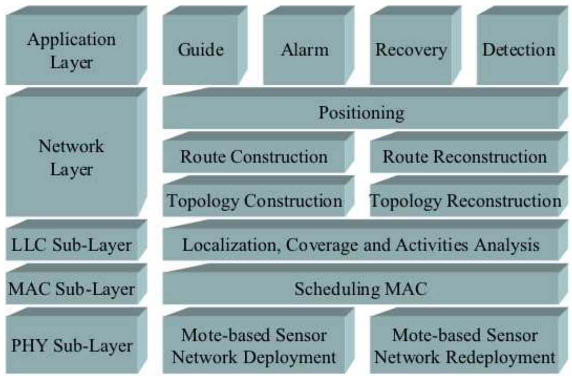


Fig. 3. Protocol stack of the GUARD system.

in the campus, sensor nodes can sense the moving object by their sensing components. These moving objects should be viewed as obstacles to the blind. When a obstacle is sensing by a sensor node or set by a user, the GUARD system can position the object through the positioning algorithm. Afterward the GUARD system will immediately **alarm** the blind who is near the obstacle. Consequently, the blind can keep away from the obstacles. The blind should be guide to another path when necessary.

- Since sensor nodes are not rechargeable, some of the sensor nodes will failed after a period of working time. The GUARD system can dynamically detect the failure of sensor nodes. The GUARD system will reconstruct the topology of the WSN deployed in Tamkang University if it finds that some of sensor nodes fail. Consequently, the walking path of the blind will be recalculated when the topology is changed. The GUARD system will use the Topology reconstruction algorithm in Network layer to reconstruct the topology of the WSN. After the topology is reconstructed, all the routing paths will be rerouted and all the walking paths of the blind will be recalculated. Hence, all activities in the GUARD system can get **recovery**.
- The detection part in the GUARD system is used to detect the blind, the obstacles, and the failures of sensor nodes. It replies on the sensing components of sensor nodes. For this reason, GUARD can communicate with the blind through wireless communication. The GUARD system will react to what sensor nodes sense. The positions of obstacles and the blind will immediately shown on the GUI of the GUARD system. Therefore the GUARD system can **detect** obstacles and the blind through sensing components and wireless communications and trigger off other algorithms in order to serve the blind seamlessly.

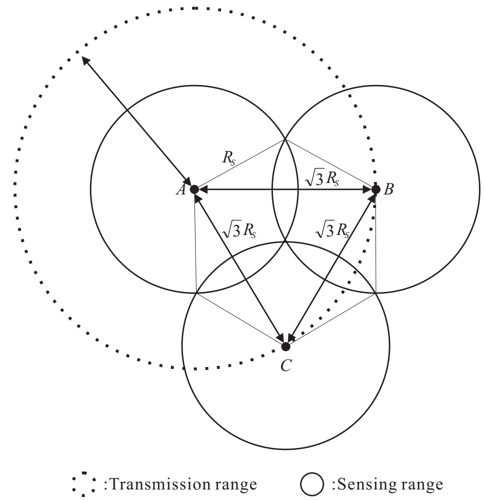


Fig. 4. The distance between sensor nodes is $\sqrt{3}R_s$. The coverage area can be maximized with the least sensor nodes.

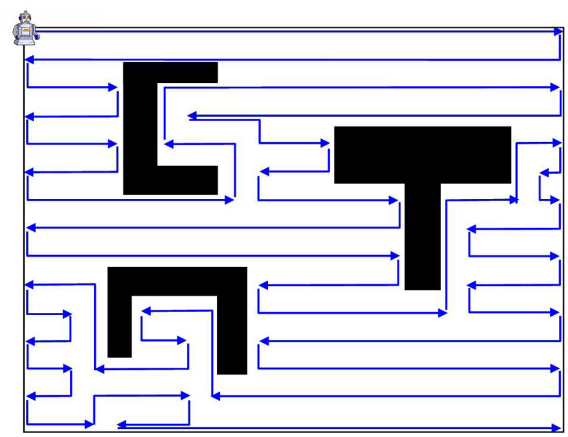


Fig. 5. A robot can deploy sensor nodes in the snake-like way.

III. Deployment by Robots

The throughput of WSNs is highly depended on the coverage status of sensor nodes. In order to achieve full coverage and save the hardware cost of sensor nodes, it's necessary to decrease the overlap of sensing ranges as much as possible. Fig. 4 shows the ranges with best coverage area. Let R_s and R_c are the sensing and transmission ranges of sensor nodes, respectively. Sensor nodes will have the maximum coverage if $R_c \geq \sqrt{3}R_s$. However, sensor nodes A still can communicate with sensor node B if the distance between A and B is 3Rs. We use this relation to design the robot deployment algorithm in our

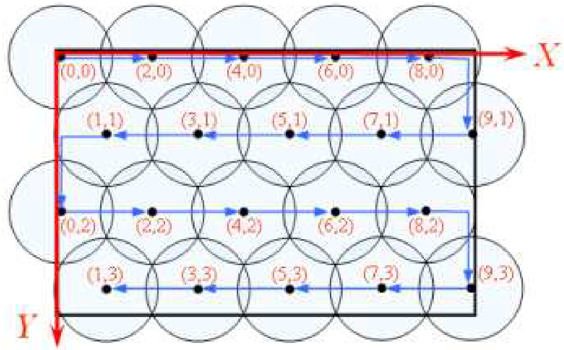


Fig. 6. The distance between sensor nodes is $\sqrt{3}R_s$. The coverage area can be maximized with the least sensor nodes.

GUARD system. In the GUARD system, robots walk in the snake-like way to deploy sensor nodes. Robots will begins from the edge of the sensing region and put sensor nodes in turns. While robots are putting sensor nodes, they also set coordinate for every sensor node. Fig. 6 shows the coordination system for WSNs. The coordination system is a two-dimensional system. The coordinate of the starting point is (0,0). Robots put sensor nodes in the snake-like way and set coordinate for every sensor nodes. Therefore, in the sensing region with obstacles, robots still can deploy sensor nodes and keep away from the obstacles. Fig. 5 shows that robots can circle the obstacles by the snake-like algorithm we designed. Consequently, every area in the sensing region can be covered by sensor nodes.

IV. Implementation of the GUARD system

A. Sensor Nodes

We will connect MICA2 Mote [9] modules to one or more personal computers in order to play the role of sinks. Fig. 7 shows the sink we constructed. We will use MICA2 Mote modules as the sensor nodes. MICA2 Mote can also play the part of mobile nodes when connecting to autonomous vehicle.

When a sensor node sensing a environment information, it can transmit this information to MICA2 Mote through analog to digital converter (ADC). Fig. 8 shows the basic component of a sensor node. The basic component of a snesor node is composed of Micro Control Unit (MCU), RF module, sensing component, power unit and I/O interface. MCU is a micro processor. It can compute and process the received and sensing data according to the need of applications. RF module is the wireless communication module. Sensor nodes can communicate with others

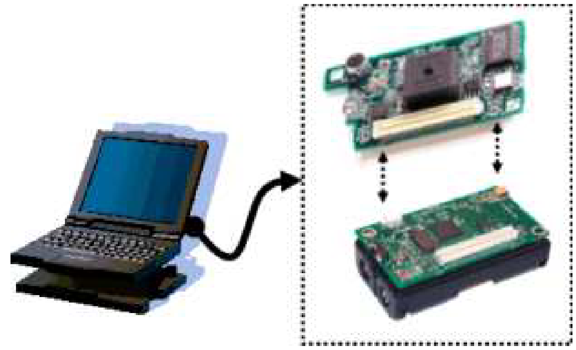


Fig. 7. A sink is composed of a notebook and a MICA2 Mote.

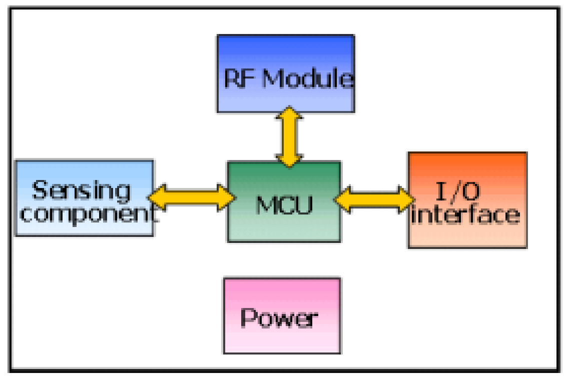


Fig. 8. Basic components of a sensor node.

through RF Module. Sensing Component is used for sensing environment information. Through different sensing components, sensor nodes can get different information from environment. I/O unit is the basic input and out device in sensor nodes. Power unit supplies energy for all other components in sensor nodes.

B. Mobile Sensor Nodes

In addition to use MICA2 Motes as sensor nodes, we combine a MICA2 Mote and a autonomous vehicles as a mobile sensor. Mobile sensors can move within some region in order to substitute for common sensor nodes. Mobile sensors can also patrol to find the damage sensor nodes.

Components of a mobile sensor are ultra sonic module, MICA2 Mote, MICA2 Mote connector, motor control integrated circuit, motor, power unit, and location generator. Fig. 9 shows the components of a mobile sensor. A mobile sensor will use ultra sonic module to detect the existence of obstacles and to determine the distance between an obstacle and itself. These information will

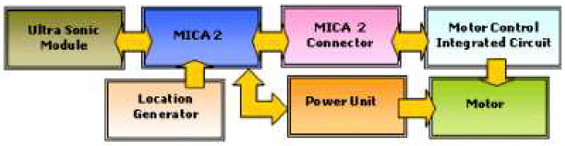


Fig. 9. Components of a mobile sensor.

```
// Blink.nc
Configuration Blink {

              }
  implementation {

    components Main, BlinkM, SingleTimer, LedsC;

    Main.StdControl -> BlinkM.StdControl;
    Main.StdControl -> SingleTimer.StdControl;
    BlinkM.Timer -> SingleTimer.Timer;
    BlinkM.Leds -> LedsC;
}
```

Fig. 10. An example of Configuration program.

```
//BlinkM.nc
module BlinkM {
  provides {
                  interface StdControl;
  }
  uses {
                  interface Timer;
                  interface Leds;
  }
}
implement{
  command result_t StdControl.init() {
      call Leds.init();
      return SUCCESS;
  }
  command result_t StdControl.start() {
      return call
Timer.start(TIMER_REPEAT, 1000) ;
  }
  command result_t StdControl.stop() {
   return call Timer.stop();
  }
  event result_t Timer.fired() {
      call Leds.redToggle();
      return SUCCESS;
  }
}
```

Fig. 11. A fragment of Module program

be sent to MICA2 Mote. A MICA2 Mote is the data processing center of a mobile sensor. All received data and sensing information will be processed in the MICA2 Mote. RF module can be used to transmit processed data and sensing information to neighboring static sensor nodes. MICA2 connector is the interface between MICA2 and motor control integrated circuit. MICA2 can control motor control integrated circuit through MICA2 connector in order to control the operation of motor. Location generator is used to know the position of a mobile sensor. We can know the position of a autonomous vehicle through location generator.

C. Communication module

Software of MICA2 Mote is developed in TinyOS [10] by University of California, Berkeley. TinyOS is a small component-based operating system. Instead of managing the processes, TinyOS uses a Scheduler to manage tasks and events. Virtual memory is not available in TinyOS, too. TinyOS configures memory statically. Therefore, TinyOS can reduce the energy consumption efficiently. We use nesC to write programs for TinyOS. NesC is a component-based and event-driven program language. The components we design include two parts. The first part is **Modules**. They can be used as the basic codes of applications. The second part is **Configurations**. They can be used to define the relations between every components. Fig. 10 shows an example of configuration program.

After we finished the Configuration program, we begin to write the Module programs. We have to check what interfaces Modules declare and what functions these interfaces define before we write the Modules. In Fig. 11, we implement three commands: init(), start(), and stop(). Due to Module supply "Std Control" interface, we can implement there commands.

Here we describe some components we design for the GUARD system.

- IntToRfm: IntToRfm is a component which can receive output value from the I/O interface. IntToRfm can also transmit the output value to other components in MICA2 Mote.
- RfmToInt: RfmToInt is a component which is used to receive packets from the radio module. After a packet is received, Mica2 Mote will check the command in the packet. If the packet is designated for other Mica2 Mote, it will be sent immediately. Otherwise, this packet will be sent to components through RfmToInt module.
- Broadcast Module: When a MICA2 Mote receive a packet, it will check the command in the packet. If the packet was received once, the MICA2 Mote will do nothing. Otherwise, the MICA2 Mote will broadcast this packet to other MICA2 Motes by Broadcast Module.
- Environment Parameter Module: When the sensing component sensing the variation of environment, Environment Parameter Module will transfer the analog signals into digital values. Then the digital values can be sent to the the GUARD system to verify the change of environments.

V. Conclusions

The system for the blind in Tamkang University had been built for years. We have had lots of experience at the using of computers for the blind, the construction of guide bricks, and the using of guide dogs. As the advance in technology, small volume, low cost, low power, and multi-functionalities wireless sensor nodes can be implemented. Wireless sensor nodes are also adopted by many applications. Consequently, we construct the GUARD system with wireless sensor nodes in order to provide security, real-time, and variety services for the blind. The capabilities of the GUARD system compose of guild, alarm, recovery, and detection. No matter how we construct the wireless sensor network, we can use the least sensor nodes to fully cover the whole region. Besides, we can extend the lifetime of WSNs through our achievement. We also analyze the relations between sensing range and transmission range of sensor nodes. Deploying sensor nodes by robots is also studied. Therefore we can deploy sensor nodes in Tamkang University by robots without blocking by obstacles. Because off-the-shelf sensor nodes do not meet our requirements, we design and implement sensing and communication modules in present sensor nodes. In order to build the GUARD system for Tamkang University, we survey the buildings and paths in order to implement the GUARD system according to the realistic scenario in Tamkang University. Through the GUARD system, positions of the blind can be shown in the GUI system. The blind can be guided to their destination properly and quickly.

VI. Acknowledgement

The authors would like to thank the National Science Council of the Republic of China, Taiwan for supporting this research under grants of NSC 93-2745-E-032-005-URD and NSC 94-2745-E-032-005-URD.

References

[1] A. Howard, M.-J. Marari'c, and G.-S. Sukhatme, "An incremental self-deployment algorithm for mobile sensor networks," *Autonomous Robots*, vol. 13, pp. 113–126, 2002.

[2] L. Li, J.-Y. Halpern, P. Bahl, Y.-M. Wang, and R. Wattenhofer, "Analysis of a cone-based distributed topology control for wireless multi-hop networks," in *Proceedings of the twentieth annual ACM symposium on Principles of distributed computing*, 2001.

[3] F. Mondinelli and Z. K. Vajna, "Self localizing sensor network architectures," in *Proceeding of the 19th IEEE Instrumentation and Measurement Technology Conference*, 2002.

[4] E. L. Lloyd, R. Liu, M. V. Marathe, R. Ramanathan, and S. S. Ravi, "Algorithmic aspects of topology control problems for ad hoc networks," in *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking and computing*, 2002.

[5] H. Gupta, S.-R. Das, and Q. Gu, "Connected sensor cover: Self-organization of sensornetworks for efficient query execution," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking and computing*, 2003.

[6] G. Gupta and M. Younis, "Load-balanced clustering of wireless sensor networks," in *IEEE International Conference on Communications*, 2003.

[7] W.-R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, 2000.

[8] The official macromedia website. [Online]. Available: http://www.macromedia.com

[9] *Wireless Measurement System, MICA2*, Crossbow Technology, Inc.41 Daggett Dr.San Jose, CA 95134.

[10] TinyOS. [Online]. Available: http://webs.cs.berkeley.edu/tos/

Proceedings

The 2005 International Workshop on Distance Education Technologies (DET 2005)

Editor

Lawrence Y. Deng, St. John's & St. Mary's Institute of Technology, Taiwan

A User-Oriented Query Prediction and Cache Technique for FAQ Proxy Service

Sheng-Yuan Yang
Dept. of Computer and Communication Engineering,
St. John's and St. Mary's Institute of Technology;
Dept. of Electronic Engineering, National Taiwan University of Science and Technology, TAIWAN

Pen-Chin Liao
Dept. of Electronic Engineering, National Taiwan University of Science and Technology, TAIWAN

Cheng-Seen Ho
Department of Computer Science and Information Engineering, Chung Kuo Institute of Technology;
Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, TAIWAN

Abstract

This paper discusses how an improved sequential pattern mining technique helps query prediction and query cache in a FAQ system. The proposed technique helps construct an FAQ proxy service, which can provide user-oriented query service as well as reduce the query loading of the backend process of a FAQ system. It can discover the users' query behavior from their query histories and base on that to construct Cache Pool and Prediction Model, which can support efficient query prediction and cache services for reducing query response time. Our experiment shows 12.2% queries can be taken care of by the query prediction mechanism, while 19.1% queries can be taken care of by the query cache mechanism. In summary, around 31.3% of the user queries can be answered by the proposed FAQ proxy service, leaving about 68.7% of the queries for the backend process to take care, which can effectively alleviate the overloading problem usually associated with a backend server.

Keywords: Query prediction, Query cache, Ontology, FAQ proxy services

1. Introduction

With increasing popularity of the Internet, people depend more on the Web to obtain their information. The use of the World Wide Web has been leading to a large increase in the number of people who access FAQ knowledge bases to find answers to their questions [14]. However, the basic operational pattern of all general Web query systems is to pass the user's query to a backend process, which is responsible for producing proper query result for the user. One major drawback of this approach is, when the number of queries increases, the backend process is overloaded, causing dramatic degradation of system performance. The user then has to spend more time waiting for query responses. Worse than that, most of the long-awaited responses are usually far from satisfactory. Therefore, how to fast get the information the users really want from the limited bandwidth of the Internet is becoming an important research topic.

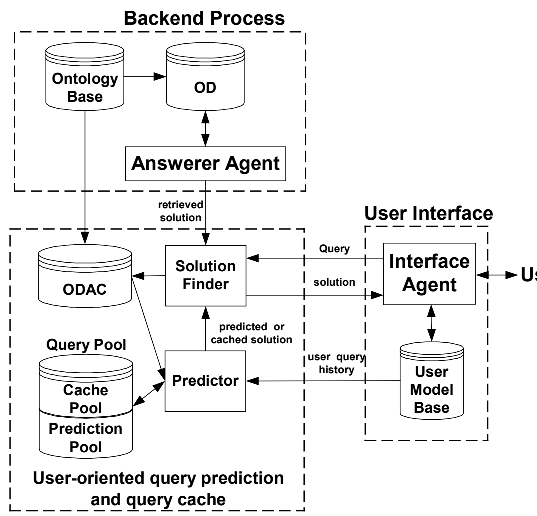


Fig. 1 System architecture

In this paper, we propose a user-oriented query prediction and cache technique for FAQ proxy service to reduce query response time. Fig. 1 illustrates its architecture and shows how it interacts with User Interface and Backend Process. First, Interface Agent collects queries from the user according to his user model [17,19]. Answerer Agent [18] serves as a back-end process. It manages OD (Ontological Database), which stores pre-processed Q-A pairs collected from the Web, and retrieves proper Q-A pairs from OD for response to user queries. Ontological Database Access Cases (ODAC) stores representative past user query cases to support query cache and prediction. The sources of ODAC are the new cases created by Answerer Agent. Predictor is responsible for tracking user query history stored in User Model Base and identifying most often occurring queries for storage in Cache Pool; it also predicts next possible queries for storage in Prediction Pool. Solution Finder is in charge of finding solutions by first invoking Predictor to retrieve predicted or cached solutions for a given user query. Specifically, Predictor first checks for any possible predicted queries in Prediction Pool. If none exists, it turns to check whether any cached queries exist in Cache Pool. If still nothing is returned from Predictor, Solution Finder will finally pass the query to Backend Process to produce a solution from OD.

With these two pools, together called Query Pool, we expect Predictor can support efficient FAQ proxy to reduce query response time [6,15]. Our experiment indeed shows 12.2% queries can be taken care of by the query prediction mechanism, while 19.1% queries can be taken care of by the query cache mechanism. In summary, around 31.3% of the user queries can be answered by the query prediction and cache technique, leaving about 68.7% of the queries for the backend process to take care, which can effectively alleviate the overloading problem usually associated with a backend server. The Personal Computer (PC) domain is chosen as the target application of the proposed system and will be used for explanation in the remaining sections.

2. Domain Ontology as Common Semantics and Supports Fast Database Retrieval

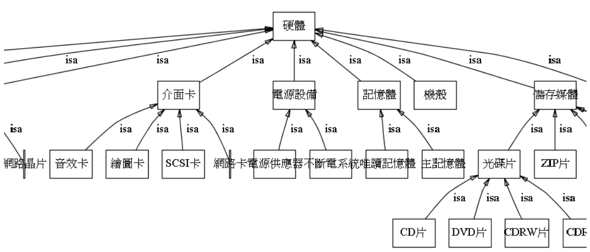


Fig. 2 Part of PC ontology taxonomy

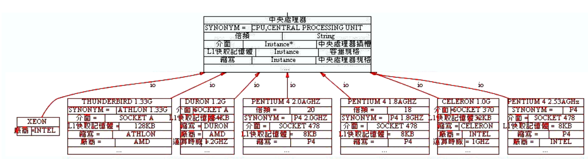


Fig. 3 Ontology for the concept of “中央處理器” (CPU)

Domain ontology is used in the system as the common semantics for FAQs. First, the ontology for the PC domain was developed using Protégé 2000 [9]. Fig. 2 shows part of the ontology taxonomy. The taxonomy represents relevant PC concepts as classes and their relationships as *isa* links, which allows inheritance of features from parent classes to child classes. We have carefully selected, from each concept, the properties that are most related to our application and defined them as the detailed ontology for the corresponding class. Fig. 3 exemplifies the detailed ontology for the concept “中央處理器” (CPU). In the figure, the uppermost node uses various fields to define the semantics of the CPU class, each field representing an attribute of “CPU”, e.g., interface, provider, synonym, etc. The nodes at the lower level represent various CPU instances, which capture real world data. The complete PC ontology can be referenced from Protégé Ontology Library at Stanford Website (http://protege.stanford.edu/ontologies/ontologies.html).

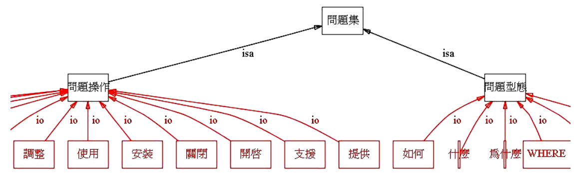


Fig. 4 Part of problem ontology taxonomy

We have also developed a problem ontology to deal with query questions. Fig. 4 illustrates part of the Problem ontology, which contains “問題型態” (question type) and “問題操作” (question operation). These two concepts constitute the basic semantics of a user query and are therefore used as indices to structure the solutions in OD and ODAC so that they can support fast solution retrieval. Finally, We use Protégé’s APIs (Application Program Interface) to develop a set of ontology services, which work as the primitive functions to support the application of the ontologies. The ontology services currently available include transforming query terms into canonical ontology terms, finding definitions of specific terms in ontology, finding relationships among terms, finding compatible and/or conflicting terms against a specific term, etc.

3. Query Cache and Query Prediction Techniques

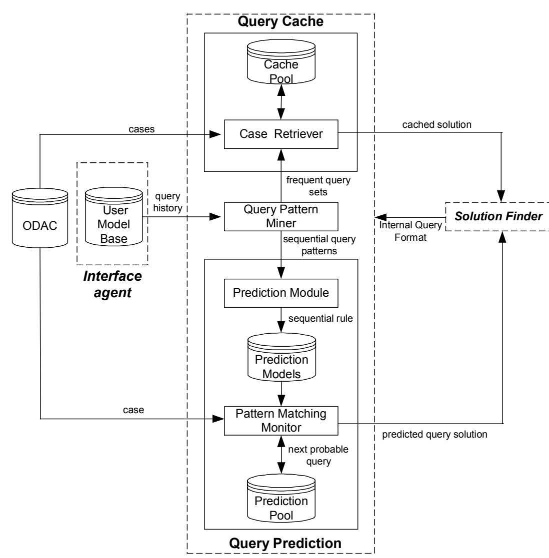


Fig. 5 Detailed architecture of Predictor

The basic idea behind Predictor is the observation of common sequential patterns in the query behavior of the users. Predictor exploits this observation and uses the sequential pattern mining technique to analyze implicit correlationships from the user query behavior. These correlationships then serve as potential rules for prediction of future user query. Fig. 5 shows the detailed architecture. First, Query Pattern Miner looks for frequent sequential query patterns inside each user group, using the Full-Scan-with-PHP algorithm, from the query histories of the users of the same group, as recorded in the User Models Base [6,15]. Note that we pre-partitioned the users into five user groups according to their proficiency on the domain [19]. Query Miner then turns the frequent sequential query patterns to Case Retriever, which is responsible for retrieving corresponding solutions from ODAC and constructing "frequent queries" for storage in Cache Pool. Prediction Module finally bases on the frequent sequential query patterns to construct a prediction model for each user group. Pattern Matching Monitor is responsible for monitoring recent query records and using the prediction model to produce next possible queries for storage in Prediction Pool. In summary, on the off-line operation, Predictor is used to produce "frequent queries" for Cache Pool and "predicted queries" for Prediction Pool. During on-line operation, given a new query, Solution Finder passes the query to Predictor, which employs both query prediction and query cache mechanisms for producing possible solutions for the query.

3.1 Full-Scan-with-PHP Algorithm

```
Input : dataset :D1, minimum support threshold :min_supp
output: all frequent patterns
          // map all patterns with perfect hashing
1: map all 1-patterns of transaction of D1 to perfect hash table
PH1;
2: L1 = { h| h∈ PH1, h.bucket_counter >= min_supp};
3: k =2;
   Do {
4:      Dk ={ };
5:      forall transaction t∈ Dk-1
           {     // perfect hashing
6:           map all k-patterns of t to perfect hash table PHk;
7:           t' =prune t , Dk =Dk∪{ t' };
           }
8:      Lk = {h|h ∈ PHk, h.bucket_counter >= min_supp};
9:      k=k+1;
     } while(Lk-1 !=<NULL>)
10: L=L1∪L2∪...∪Lk;
11: return L;
```

Fig. 6 Full-Scan-with-PHP algorithm

The Full Scan algorithm was based on the DHP algorithm to perform sequential pattern mining [3]. The PHP algorithm introduced the perfect hash mechanism into the DHP algorithm and proved it performs better than the DHP algorithm [10]. Fig. 6 integrates PHP with Full Scan to form our sequential pattern mining algorithm, which returns L (step 10) as a set of frequent sequential patterns. The pruning method (step 7) is the same as PHP. The major feature of the algorithm is that it can obtain L_k by directly scanning the perfect hash table PH_k (step 8) without spending extra time scanning the database to calculate supports for the frequent sequent patterns. Fig. 7 illustrates how the algorithm works on example database D_1 with minimum support *min_supp* set to 2.

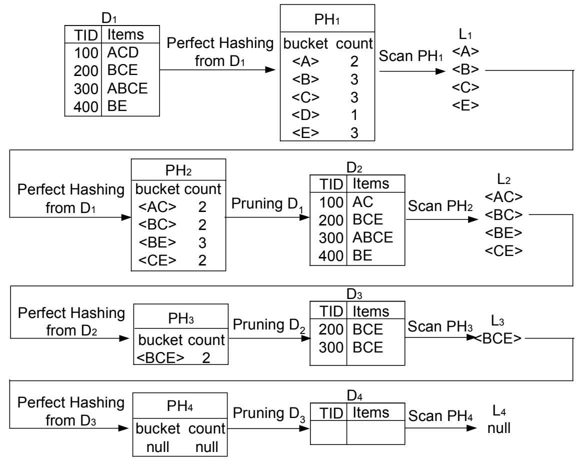
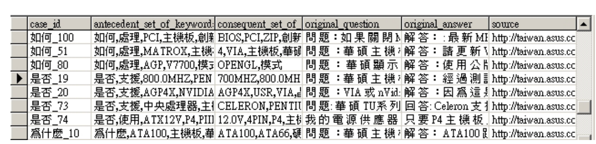


Fig. 7 Illustration on sequential pattern mining of Full-Scan-with-PHP algorithm

3.2 Construction of Cache Pool

As noted before, Cache Pool contains frequent queries while Prediction Pool contains predicted queries. Both are constructed based on the frequent sequential query patterns discovered by the Full Scan with PHP algorithm. Note that we treat each query case stored in ODAC as an item during the operation of the algorithm. The algorithm first constructs L_1 as a set of 1-item frequent sequential patterns. Strictly speaking, this set has nothing to do with query sequences yet, but it represents the pool of all frequent queries. Thus, each element in the set can be treated as a frequently happening query, which can be cached somewhere for ready use in the future. We hence hand each element of the set to Cache Retriever, which then retrieves its corresponding answer part from ODAC to form a complete frequent query for storage in Cache Pool. Fig. 8 illustrates some examples of frequent queries stored in the Cache Pool.



case_id	antecedent_set_of_keyword:	consequent_set_of_	original_question	original_answer	source
如何_100	如何,處理,PCI,主機板,創	BIOS,PCI,ZIP,創新	問題：如果關閉	解答：：最新	http://taiwan.asus.cc
如何_51	如何,處理,MATROX,主機	4,VIA,主機板,華碩	問題：華碩主機	解答：請更新	http://taiwan.asus.cc
如何_80	如何,處理,AGP,V7700,模	OPENGL,模式	問題：華碩顯示	解答：使用公	http://taiwan.asus.cc
是否_19	是否,支援,800.0MHZ,PEN	700MHZ,800.0MH	問題：華碩主機	解答：經過測	http://taiwan.asus.cc
是否_20	是否,支援,AGP4X,NVIDIA	AGP4X,USR,VIA,	問題：VIA 或 nVid	解答：因爲這	http://taiwan.asus.cc
是否_73	是否,支援,中央處理器,主	CELERON,PENTIU	問題: 華碩 TU系列	回答: Celeron 支	http://taiwan.asus.cc
是否_74	是否,使用,ATX12V,P4,PII	12.0V,4PIN,P4,主	我的電源供應器	只要 P4 主機板	http://taiwan.asus.cc
爲什麼_10	爲什麼,ATA100,主機板,華	ATA100,ATA66,硬	問題：華碩主機	解答：ATA100	http://taiwan.asus.cc

Fig. 8 Examples of frequent queries in Cache Pool

3.3 Construction of Prediction Pool

The final output of the mining algorithm returns L a set of frequent sequential patterns, which is submitted to Prediction Module for producing sequential rules. The basic idea follows [20] by defining the sequential rule confidence as follows.

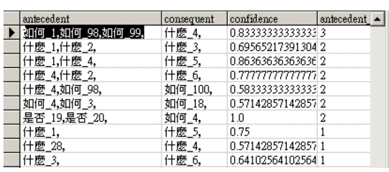
Definition 1: Given two frequent sequential patterns S_1=<$t_1,t_2,\ldots,t_k$> and S_2=<$t_1,t_2,\ldots,t_{k-1}$>, where $k \geq 2$ and S_2 is a sub-sequential pattern of S_1, we produce a sequential rule "$t_1,t_2,\ldots,t_{k-1} \rightarrow t_k$" with the following rule confidence:

$$\text{Confidence}(t_1,t_2,\ldots,t_{k-1} \rightarrow t_k) = \text{Support}(S_1) / \text{Support}(S_2)$$

Now we define that a sequential rule is legal if its confidence satisfies some minimal confidence. Table 1 shows part of the legal sequential rules generated from the database D_1 of Fig. 7 with minimal confidence 0.6. Fig. 9 illustrates some examples of prediction rules about the PC domain stored in the prediction model.

Table 1 Examples of sequential rules

Sequential Rule	Confidence
B,C → E	1
A → C	1
B → C	0.67
B → E	1
C → E	0.67



	antecedent	consequent	confidence	antecedent_
▶	如何_1,如何_98,如何_99,	什麼_4,	0.83333333333333	3
	什麼_1,什麼_2,	什麼_3,	0.69565217391304	2
	什麼_1,什麼_4,	什麼_5,	0.86363636363636	2
	什麼_4,什麼_2,	什麼_6,	0.77777777777777	2
	什麼_4,如何_98,	如何_100,	0.58333333333333	2
	如何_4,如何_3,	如何_18,	0.57142857142857	2
	是否_19,是否_20,	如何_4,	1.0	2
	什麼_1,	什麼_5,	0.75	1
	什麼_28,	什麼_4,	0.57142857142857	1
	什麼_3,	什麼_6,	0.64102564102564	1

Fig. 9 Examples of rules in the prediction model

With the prediction model constructed, Pattern Matching Monitor can do query prediction and produce predicted queries into Prediction Pool. Basically, it monitors and traces the "query record" of the user in a session, and employs the longest-path-first method [20] to check whether any matched sequential rules exist in the prediction model. A query record is as sequence of queries. If the antecedent of a sequential rule matches the pattern of the traced query record, Pattern Matching Monitor will predict the consequent of the rule as the next possible query and retrieves its corresponding solution part from ODAC to form a predicted query for Prediction Pool. If no rules exist, the Monitor will decrease one query from the query record and repeat the sequential rule checking process until the length of the query record reaches one. Let's illustrate this process of query prediction below, which uses the prediction model of Table 1. Suppose Pattern Matching Monitor notices a user query record <A, B, C> in a session. It checks the prediction model and finds no antecedents of sequential rules match <A, B, C>. Therefore, it reduces the traced pattern into <B, C> and repeats the rule checking process. It finally finds the antecedent of sequential rule "B,C → E" matches the new pattern. It thus predicts the consequent "E" as next query and retrieves its answer part from ODAC to form a predicted query.

4. System Evaluation

The system was developed using Borland JBuilder 5.0 on Microsoft Windows XP. The database management system is Microsoft SQL Server 2000 and ontology development tool is Protégé2000. We collected in total 517 FAQs from the FAQ website of one famous motherboard factory in Taiwan and then transformed the query-answer pair of each FAQ for storage in the ODAC. Our experiment was to learn how well Predictor works. We used in total 200 user query scenarios of the same user level as the training data set. We set the minimal support to 3% and minimal confidence to 60%. In the experiment, the Full-Scan-with-PHP algorithm constructed 36 frequent queries for storage in Cache Pool and 43 rules in Prediction Model. Prediction Pool only kept the most recent three predicted queries, in order to avoid over-fitted prediction, which produces a solution similarity to rote learning of lots of prefetched queries. We then randomly selected 100 query scenarios from the training data set as the testing data to test the performance of Predictor. Table 2 illustrates the five-time experiment results. It shows, on average, Query Prediction is invoked to predict next query for 60.2 queries, among which it correctly predicted next query for 38 queries, a 65.8% average success rate. It also shows, on average, out of 310 queries, 38 queries can be taken care of by the query prediction mechanism (around 12.2%), while 59.2 queries can be taken care of by the query cache mechanism (around 19.1%). In summary, around 31.3% of the user queries can be answered by the user-oriented query prediction and cache technique.

Table 2 Testing results on Predictor

Testing Order	#Query	#Query Prediction Invocation	Query Prediction		Query Prediction Success rate	Query Cache	
			#	%		#	%
1	289	41	27	9.3 %	65.8 %	52	18.0 %
2	325	88	44	13.5 %	50 %	72	22.1 %
3	320	59	47	14.7 %	79 %	58	18.1 %
4	302	65	39	12.9 %	66.1 %	59	19.5 %
5	314	48	33	10.5 %	68 %	55	17.5 %
Average	310	60.2	38	12.2 %	65.8 %	59.2	19.1 %

5. Related Works

User action prediction is a popular subject. For example, the Letizia system [7] predicts user navigation actions using forward explorations. The Syskill & Webert system [11] learns to rate pages on the Web. The Web Watcher system [4] anticipates the next selected hyperlink by using a model built through reinforcement learning. The Transparent Search Engine system [2] evaluates the most suitable documents in a repository using a user model updated in real time. An alternative approach to Web pages prediction is based on "Path". For example, the work by Zuckerman et al.

[21] uses a web server log as training information to learn a Markov model, and the predicted documents by the model are then sent to a cache located on the client side. Lau and Horvitz [5] has characterized user queries with a Bayesian model in order to make predictions on users' next query based on previous queries and time intervals. Pitkow and Pirolli [12] proposes a Longest Repeating Sequences approximation of Kth-order Markov models using N-gram representation of user request that extracts user patterns from Web log data. WhatNext [13] obtains a model by building sequences of user requests of sufficient length from a web server log file, and predicts the next user action by exploiting a statistical analysis of sequence information. The work of Yang, Zhang and Li [16] presents an application of Web log mining to obtain web-document access patterns and use these patterns to extend the well-known GDSF (Greedy-Dual-Size-Frequency) caching policies and prefetching policies. PPS (Proxy-based Prediction Service) [8] applies a new prediction scheme which employs a two-layer navigation model to capture both inter-site and intra-site access patterns, incorporated with a bottom-up prediction mechanism which exploits reference locality in proxy logs. The work of Bonino, Corno and Squillero [1] proposes a new method to exploit user navigational path behavior to predict, in real-time, future requests using the adoption of a predictive user model based on Finite State Machines (FSMs) together with an evolutionary algorithm that evolves a population of FSMs for achieving a good prediction rate. In comparison, our work adopts the technique of sequential patterns mining to discover user query behavior from the query history and accordingly offer efficient query prediction and query cache services.

6. Conclusions

The proposed user-oriented query prediction and cache technique can efficiently predict and cache proper solutions for a given user query to relieve the processing loading of a backend FAQ server with the help of an improved sequential pattern mining technique and domain ontology. It is interesting at the following facets. First, it performs fast user-oriented mining and prediction. Predictor uses the user query history stored in User Model Base to discover frequent queries and predicted queries and bases on that to construct Cache Pool and Prediction Model, which can support efficient query prediction and query cache services. The sequential pattern mining algorithm is made more efficient by the techniques of perfect hashing and database decomposition. Finally, our experiment shows 12.2% queries can be taken care of by the query prediction mechanism, while 19.1% queries can be taken care of by the query cache mechanism. In summary, around 31.3% of the user queries can be answered by the user-oriented query prediction and cache technique, leaving about 68.7% of the queries for the backend process to take care, which can effectively alleviate the overloading problem usually associated with a backend server.

Acknowledgements

The authors would like to thank Ying-Hao Chiu, Yai-Hui Chang, and Fang-Chen Chuang for their assistance in system implementation. This work was supported by the National Science Council, R.O.C., under Grants NSC-89-2213-E-011-059 and NSC-89-2218-E-011-014.

References

[1] D. Bonino, F. Corno, and G. Squillero, "A Real-Time Evolutionary Algorithm for Web Prediction," *Proc. of the 2003 IEEE/WIC International Conference on Web Intelligence (WI'03)*, Halifax, Canada, 2003, 139-145.

[2] F. Bota, F. Corno, L. Farinetti, and G. Squillero, "A Transparent Search Agent for Closed Collections," *International Conference on Advances in Infrastructure for e-Business, e-Education, e-Service, and e-Medicine on the Internet*, L'Aquila, Italy, 2002, 205-210.

[3] M.S. Chen, J.S. Park, and P.S. Yu, "Efficient Data Mining for Path Traversal Patterns," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 10, No. 2, 1998, 209-221.

[4] T. Joachims, D. Freitag, and T. Mitchell, "WebWatcher: A Tour Guide for the World Wide Web," *Proc. of the 1997 International Joint Conference on Artificial Intelligence (IJCAI'97)*, Nagoya, Japan, 1997, 770-775.

[5] T. Lau and E. Horvitz, "Patterns of Search: Analyzing and Modeling Web Query Refinement," *Proc. of the 7th International Conference on User Modeling (UM'99)*, Banff Centre, Banff, Canada, 1999, 119-128.

[6] P.C. Liao, An Intelligent Proxy Agent for FAQ, Master thesis, Department of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, 2003.

[7] H. Lieberman, "Letizia: An Agent that Assist Web Browsing," *Proc. of the 1995 International Joint Conference on Artificial Intelligence (IJCAI'95)*, Montreal, Canada, 1995, 924-929.

[8] W.W. Lou and H.J. Lu, "Efficient Prediction of Web Accesses on a Proxy Server," *Proc. of the 11th International Conference on Information and Knowledge Management (CIKM'02)*, McLean, Virginia, USA, 2002, 169-176.

[9] N.F. Noy and D. L. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology," Available at http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness.pdf, 2000.

[10] S.A. Özel and H.A. Güvenir, "An Algorithm for Mining Association Rules Using Perfect Hashing and Database Pruning," *Proc. of the 10th Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN'01)*, North Cyprus, 2001, 257-264.

[11] M. Pazzani, J. Muramatsu, and D. Billsus, "Syskill & Webert: Identifying Interesting Web Sites," *Proc. of the 13th National Conference on Artificial Intelligence (AAAI'96),* Portland, Oregon, USA, 1996, 54-61.

[12] P. Pirolli and J. Pitkow, "Distributions of Surfers' Paths through the World Wide Web: Empirical Characterization," *World Wide Web*, Vol. 2, No. 1-2, 1999, 29-45.

[13] Z. Su, Q. Yang, Y. Lu, and H. Zhang, "WhatNext: A Prediction System for Web Requests Using N-gram Sequence Models," *Proc. of the 1st International Conference on Web Information System and Engineering Conference (WISE'00)*, Hong Kong, China, 2000, 200-207.

[14] W. Winiwarter, "Adaptive Natural Language Interface to FAQ Knowledge Bases," *International Journal on Data and Knowledge Engineering*, Vol. 35, 2000, 181-199.

[15] C.W. Wu, Using Rule Mining and Behavior Prediction Techniques in Webpages Query Processing, Master Thesis, Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, 2001.

[16] Q. Yang, H.H. Zhang, and T. Li, "Mining Web Logs for Prediction Models in WWW Caching and Prefetching," *Proc. of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'01)*, San Francisco, California, USA, 2001, 473-478.

[17] S.Y. Yang, Y.H. Chiu, and C.S. Ho, "Ontology-Supported and Query Template-Based User Modeling Techniques for Interface Agents," *2004 The 12th National Conference on Fuzzy Theory and Its Applications*, I-Lan, Taiwan, 2004, 181-186.

[18] S.Y. Yang and C.S. Ho, "An Intelligent Web Information Aggregation System Based upon Intelligent Retrieval, Filtering and Integration," *The 2004 International Workshop on Distance Education Technologies (DET'2004)*, Hotel Sofitel, San Francisco Bay, CA, USA, 2004, 451-456.

[19] S.Y. Yang and C.S. Ho, "Ontology-Supported User Models for Interface Agents," *Proc. of the 4th Conference on Artificial Intelligence and Applications (TAAI'99)*, Chang-Hua, Taiwan, 1999, 248-253.

[20] H. Zhang, "Improving Performance On WWW Using Path-Based Predictive Caching And Prefetching," Available at http://citeseer.nj.nec.com/459947.html, 2001.

[21] I. Zuckerman, D.W. Albrecht, and A.E. Nicholson, "Predicting User's Request on the WWW," *Proc. of the 7th International Conference on User Modeling (UM'99)*, Banff Centre, Banff, Canada, 1999, 275-284.

Use Knowledge Management Base to Establish the e-Learning Interactive Teaching Platform for the Autism Students

Huay Chang[12], Shieh-Shing Lin[3], Pei-Ru Liu[1], Kun-Lin Chung[1],
Wen-Wei Lin[1], Pei-Pin Wu[1], Tsung-Yuan Wu[1]
[1] Chihlee Institute of Technology
[3] St. John's & St. Mary's Technique University
cecilia2chang@yahoo.com.tw[1 2]

ABSTRACT

In this paper, we made an analysis on the autism students' learning platform. We designed an interactive e-learning teaching platform. In this platform, we provide the special-education teachers multi-materials in teaching area and create teaching evaluation graphs automatically. This long-distance teaching model breaks out the space limit and time limit. This model not only integrates the resources on the special education area but also provides the central management to control the sharing channels of the resources. The main characteristic of the platform is the 'sound identification learning'. This helps the autism students raising their ability in sound identification and opens another door for their learning.

Keywords: autism, e-learning, teaching platform, long distance teaching

1. INTRODUCTION

1.1 Research Background

Due to the time and space advantages, knowledge spreads out fast today. The e-learning teaching models are designed all over the world. Under the platform of the open e-learning teaching modes, there is a center of knowledge interchanging [1]-[4]. This center owns the functions of data collecting and data sharing. This center relies on the platform in seeking knowledge, processing knowledge, storing knowledge, and creating new knowledge finally [5]-[7].

Therefore, by using the 'knowledge management' base to establish e-learning education resource platform for the autism students would help in knowledge managing of the platform and establishing a channels for the autism students to contact multi-topic learning areas through the internet. We wish the autism students benefit from the hi-tech era.

1.2 Research Motive

The autism students couldn't use fluently language and characters to express themselves and to communicate with normal people [8]-[11]. Therefore, if we can apply the information techniques and communication techniques in learning and teaching process, it would bring the apparent benefits for the autism students and their teachers. The establishment of the e-learning platforms creates the new hope for those autism students.

The autism students lack the ability in quantity judge, their scholastic materials should be designed by doctors and special education teachers. Although there are many types of e-learning platforms, they are not suitable for those autism students. Therefore, establishing an e-learning platform for the students is our first research motivation.

The special education teachers wish to gather all the resources from the e-learning teaching platform. They wish to arrange the special teaching program for each autism student by using the e-learning platform. This individual teaching programs would help the autism student begins learning at anytime and anywhere. This creation on educational curing program is the second one of our research motivations.

All the resources gathered on the e-learning platform help the special education teachers in the creation of teaching programs and help the autism students in learning programs. The knowledge gathered in the e-learning platform creates new knowledge day by day. This would help in solving the teaching problems of the autism students but give a channel in analyzing the autism related problems. This is the third one of our research motivations.

1.3 Research Purpose

Following are three research purposes:

i. Establishing e-learning interactive teaching platform for the autism students and strengthen the quantity ability against the audio animation of the autism students.

ii. The special education teachers can use the e-learning interactive teaching platform to arrange the individual teaching model. And this platform would be a permanently learning model for the autism students.

iii. Importing the knowledge management model via the e-learning interactive teaching platform. It would provide the multi-advantages for the autism students and special education teachers.

2. LITERATURE REVIEW

This section makes a literature review in the following fields: knowledge management, e-learning, autism, and structure teaching.

2.1 Knowledge Management

The concept of the knowledge is very wide. It also owns the variety definition. The knowledge management represents various definitions, too.

Knowledge management is to provide the right knowledge at the right time to the right person. The purpose is to help the person to make the best decision. The definition of knowledge management means the organization completely explores and exploits the knowledge assets systematically and accurately [5]-[7]. This helps in raising the efficiency of the organizational work and reaching the maximum return on organizational investment.

Knowledge management is the model of the organization to use the formal channels to pursue useful experience, knowledge and professional ability. This would help the organization creating new ability, raising efficiency, promoting R&A work and strengthen customers' value.'

Nonaka (1994) submitted the 'SECI Model'. This model is created by knowledge. This knowledge is transformed by the inner & outer knowledge that explains the dynamically creating and maturing process, see Fig.1.

'Socialize' means to change inner to outer mode. 'Outer' means to change from inner to outer mode. This relies on language and pattern to express thinking and smart skills. 'Assemble' means to change from outer to inner and assembles the language. 'Inner' means to change from outer to inner and acquires language and pattern.

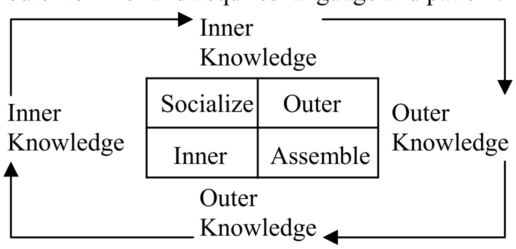


Fig. 1 SECI Model

2.2 The Discussion on e-learning

The e-learning uses electrical teaching techniques and allows learners receive specialists and instructors' knowledge without the constraints of time and spaces [2]-[3]. The e-learning includes three fields: Related learning contents and goals, using teaching methods (examples and practices) to help learning, establishing the new knowledge and skills those matches the learners' goals [8]-[10].

The three standards of the e-learning are:

i. The on-line learning inter-connected network is created. This network can update, store/access, distribute and share teaching information immediately.

ii. The on-line learning uses the standard internet network technique. It transfers data to the end user through the computers.

iii. The e-learning focuses on the wide-range learning. It exceeds the traditional learning models.

2.2.1 The Characteristics of e-learning

The characteristics of e-learning are: i. Providing equal opportunity in education and helping students receive high-quality education. ii. Eliminating the constraints and interfering on time and space and providing a convenient learning model. iii. Individual education provides flexible learning model for the students. iv. Decreasing the learning pressure for the students and emphasizing privacy. Once there are learning problems, discussion can be solved via the computers. v. The students can join learning programs with foreign students all over the world. It helps in creating the globalization view point. vi. The ability of self-control is an extra value from this learning model. It also creates the following ability: activity, responsibility, independence, and life-learning [11]-[13].

2.2.2 The Strengths of e-learning

The e-learning includes multi-type materials, the exchanging channels of learning experience, the virtual learning groups, and professional knowledge provided on network materials. It widens the information sources. It provides the rapid learning channels in this informational world. It provides the information sharing model to decrease the learning expenses. Its professional content increases the learning quality. Its learners-oriented base increases the participating rate.

i. Multi-type Models

E-learning offers the hypertext materials: such as characters, images, music, animations, and videos.

ii. Sharing Channels of Learning Experience

E-learning offers the sharing channels of learning experience. Such as, comment, discussion, chat room.

iii. Virtual Learning Group

The e-learning teaching model eliminates the constraints of time and space. Therefore, there are many learners sharing the teaching materials stored on the same teaching platform and creates the virtual learning group.

iv. Professional Providers on Network Materials

The e-learning platform gathers many specialists from different areas. Therefore, this platform stands multi-style materials and comments.

2.3 The Autism

The 'Autism' is named by the Dr. Bleuler in 1913. He found out those people who own the characteristics of extremely independence and loss the ability in communications. In 1943, Dr. Leo Kanner pointed that the eleven children whom he was analyzing own the characteristic of extremely independence are similar with the description raised by Dr. Bleuler. But, in order to distinguish those characteristic of extremely indepen- dence is not appeared accidentally. Dr. Kanner named this type of behavior 'Early Infantile Autism'. This type of disease couldn't be establish and reflected the human relations.

2.3.1 The Characteristics of Autism

The autism is a kind of behavior syndrome. It is the cluster of abnormal types. The autism will be decided only when the following behavior type appeared at the same time [14]-[17]. These behavior types are:

i. The Bafflement of Human Relations

The Autism patients lack the ability in learning how to understand themselves and how to establish the relationships with other people. These patients appear the following behavior characteristics since babyhood: scarcely talking to people, scarcely seeing people, less reaction with people, lacking fear with strangers, hard to establish close relationship with family members, lacking the imitation ability, hard to play with other children, hard to understand people's emotion, hard to express their feelings normally.

ii. The Bafflement of Language and Communication

The Autism patients own the various degree of difficulty in understanding other people's behaviors: oral communication, body language, posture, face expression. There are about 50% of the Autism children have no suitable language in communication. Even those Autism children who can speak, but they appeared the following problems: parrot-imitation speaking, pro-noun reverse, giving the wrong answer and plain tongue.

iii. The Behavior Sameness

The Autism children have different habits from other normal children. Such as, they choose the fixed route while they are walking out, they own the fixed and narrow habits in clothes style, food types, living environment and walking habit. The playing habits are very blah and needless change. The environment must be fixed, or, they can't accept and start to refuse and cry suddenly [18]-[19].

2.3.2 Autism Teaching

Structural Teaching model is developed by the North Corona University in USA. This teaching mode matches the students' development degree. The TEACCH (Treatment and Education of Autistic and Related Communication Handicapped children) is one of the most influential special education methods of the Autism students. The Structural Teaching model is appropriate for the Autism students. Its design rule is based on the sight advantage ability of the Autism students. Therefore, this Structural Teaching model uses many sight signals and hints helping the Autism students in working and learning. The Structural Teaching model also provides the Autism students the 'predictable' environment that helps them realizing their next step at the right time and the right place. This method releases their conscious [20]- [23].

3. RESEARCH METHODOLOGY

3.1 Research Framework

The research flowchart is as Fig. 2.

3.2 Research Design

Our research analyzes the relationship among knowledge management, e-learning and Autism. We also have made several interviews with the special education teachers and gained practical numeric proof on the real benefit of the e-learning work.

3.2.1 Data Collecting Method

i. Literature Analysis Method

We have read lot of books, journals, thesis, research concerning on the topics of Autism medical and e-learning technique and knowledge management. We tried to find out the relationship among the Autism students, e-learning interactive teaching method and knowledge management. Our findings indicate that the Structural Teaching model and computer single operation is acceptable. And the vision teaching used in computer matches the structural vision cue exactly. Besides, e-learning is an important information technique in knowledge broadcasting. The multi-media function of the computer also helps in the producing of the teaching materials.

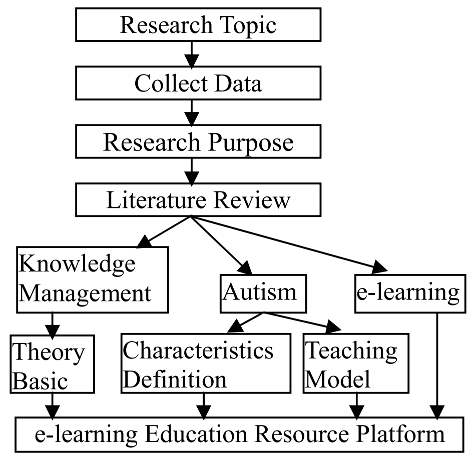


Fig. 2 Research Structure

ii. Depth Interview Method

In order to understand the reality of the Autism students learning, we have made several interviews with the special education teachers. We also invited the teachers to operate our e-learning platforms and gained much feedback and precious opinions from the teachers. These reflect a positive benefit on the support of the e-learning interactive teaching platform against the Autism students.

3.2.2 Data Analysis Result

i. The judge of the reception of the Autism students on classifying the audio combined with the animation.

90% of the special education teachers agree that teaching assistant tool that has the pictures animation would attract the attention and increase the classifying ability of the Autism students. There re are 73% of the special education teachers point out that the added audio classifying ability increases the hearing ability of the Autism students.

ii. The judge of the reasonability of the price of the existing assistant tools.

a. Over 95% of the special education teachers point out that the price of the teaching assistant tools are too high. Most of the Autism students' parents couldn't afford the high burden and limits the learning resource of the Autism students. There is about 70% of the special education teachers point out that the content of the materials is not flexible and the teaching pictures are not suitable for the Autism students. Therefore, 90% of the special education teachers develop the teaching pictures by themselves.

b. Most of the teaching assistant tools are single-computer version. Those tools can't be shared via the network. There is no such a specific software and platform designed for the special education teachers and students. Over 85% of the special education teachers wish to share the teaching resources and establish the recording function.

c. About 3.95% of the special education teachers agree that the assistant tools are designed generally and contain no special characteristics. We also find out that the shortage of the school budget limits the development of the special education assistant tools and shrinks the learning space of the Autism students.

d. For those Autism students who are classified to the low-function ability group, 90% of the special education teachers don't support that they own the ability in operating the mouse, but the track balls and touching-screen would be better suitable for those students. 95% special education teachers point out that the low-function Autism students represent the low understanding ability than other Autism students. And the common teaching assistant tools wouldn't be suitable for them, either.

iii. The tendency of the teaching materials design.

a. 90% of the special education teachers point out that most of the CAI (Computer Aided Instructions) softwares are designed for the public but not for the Autism students. All of the special education teachers think that the Autism students should turn to self-learning strongly by their personal preference. And all of the students think that the animation used in the materials should be suitable for the application of the daily life. And 95% of the special education teachers believe that, in order to catch the attraction of the Autism students, the animation used in the materials must be combined with the suitable environment.

b. Through the interviews, we found that 100% of the special education teachers agree with the high differentiation of the degree among the Autism students. And the production of the teaching assistant tools should be positioned by the individual Autism students.

c. 60% of the special education teachers emphasize that the materials should be separated into units and should be simplified. 40% of the special education teachers emphasize that the materials should be practical, the sound of the audio should be clear and the size of the picture should be big enough.

iv. The Combination Possibility of e-learning Network Platform and Audio Animation.

a. 80% special education teachers point out that through the sharing of network resource and end-learning help the autism students reaches the goal of life-learning and eliminates the limitation of time and space. 76% of the special education teachers believe that the computer network combining with multi-media matches the tendency of the era and increases the interaction relationship between the Autism students and their family members.

b. 97% of the special education teachers agree that e-learning can record the learning record of the Autism students. 90% of the special education teachers support that the usage of the e-learning platform appears a great help in future development and learning.

4. CONCLUSION AND SUGGESTION

4.1 Functional Structure of e-Learning Platform

Based on the general teaching web and combined with the requirements of the Autism students and special education teachers, we designed the functional structure of the system, as Fig.3.

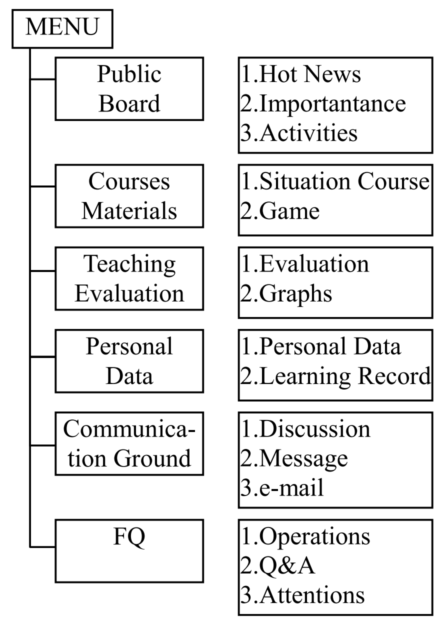


Fig. 3 Functional Structure

i. Public Board

The special education teachers put their announcement on this public board. The students and their parents gain relative information from this public board. The content contains fresh news, important issues, activities information, information updating, etc.

ii. Course and Materials

The design of the courses mainly focuses on the daily life. It uses the situational teaching model, such as the transportation unit, the park unit, etc. It also uses the computer characteristics of the multi-media. In order to increase the understanding of the daily life environment for the Autism students, this model combines with Flash animation, audio, and text description. We also adapt the game method to evaluate the learning result. The details are listed below:

a. Situational Course：

This course is designed by the one-to-one assistant teaching method. The teachers and parents accompany the Autism in operating process. The Autism students start understanding the environment around themselves by the audio and animation functions. As listed Fig. 4 & Fig. 5.

Fig. 4 Situational Course—Transportation Unit

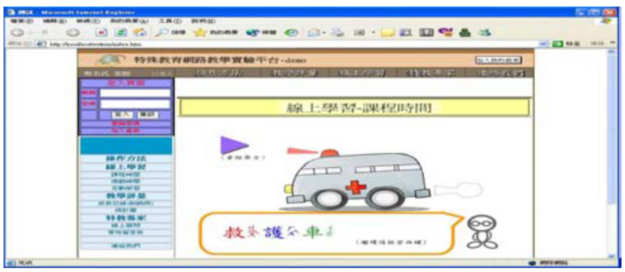


Fig. 5 Situational Course—Transportation Unit (Ambulance)

b. Game

The game decreases the fear of computers for the Autism students and increases the familiarity of the computers. Game appears a great help in the following learning courses and train Autism students the confidence and aggressive manners. In this part, the arrangements of the game content are based by the situational courses. The students can relax their mood and join the evaluation of the learning. As listed in Fig. 6.

Fig. 6 Hearing Identifying Game—Game Content

iii. Teaching Evaluation

According to the students' reaction and learning evaluation, the special education teachers can not only make detailed recordings and create statistical analytical graphs. These results help the special education teachers in analyzing and syllabus arranging. The parents can understand their children's learning result by these records and corporate with the teachers in arranging special family education for the Autism students. As listed in Fig. 7.

Fig. 7 The Analytical Chart

iv. Personal Data

This system records the Autism students' personal data and keeps their complete learning records. Those precious records provide special reference for the special education teachers at different education stages. As listed in Fig. 8.

Fig. 8 The Students' Growing Record

v. Suggestion Ground

This area includes the discussion area, message area and e-mail area. This area provides a ground for sharing opinions from medical field to Autism field. This ground also provides the two-way communication channels for the parents and special education teachers.

vi. F & Q：

This area includes operation method, important issues and limitations. This area provides detailed web-usage instruction for the teachers, Autism students and teachers.

4.2 Conclusion

In this research, we find out that the Autism students prefer the computer assisted software that owns the multi-media function. We reach the practical teaching goals through the network platform. The detailed findings are as follows:

4.2.1 The Maximum Integration of Teachers Resource

There are many special education teachers all over the world. Each special education teachers owns his (her) personal teaching method and teaching experience. Once the special education teacher uses the e-learning interactive teaching platform, his (her) teaching environment extends to the globalization teaching. The Autism students will own the maximum teachers. All the special education teachers use the platform and they devoted their maximum resources into the platform to provide the teaching resources for the Autism students.

4.2.2 The Editing of Multi-Style Materials

The special education teachers donate their teaching materials. All materials provided on the platform would decrease the iteration designing time. Due to the characteristics of the computers, the materials used on the platform can be displayed by the multi-medial types, such as text, audio, image and film, animation. Lots of researches point out that the Autism students appear special preference on computer teaching. The computer teaching attracts the attention and increases the learning of the Autism students.

4.2.3 The Arrangement of Personalize Learning

Each Autism student represents different result. Therefore, the learning arrangement and course design shouldn't be the same, either. Under the constraints of the traditional teaching model, the ratio of the teacher and students are 1:1. Most of the Autism students represent different degree in learning result. The teaching resource of the e-learning platform could become the make-up learning program for the Autism students. The special education teachers select the suitable materials and courses for each Autism student.

4.2.4 The Systematically Recording of Learning Process

While the Autism students use the e-learning platform in course learning, the system records the learning process automatically. These learning records would be very complete through the elementary school, middle- school, high-school, and college learning stages. Those records provide the teachers understanding the autism students at different education levels in advance.

4.2.5 The Automatically Creation of Teaching Evaluation

Based on the system learning records, the analytical graph can be created automatically. These graphs help the special education teachers evaluating the status of the Autism students. These graphs help the special education teachers arrange the teaching programs flexibly. These graphs also help the special education teachers arrange the individual teaching program while required.

4.2.6 The New Opportunity of the Audio-Teaching Creation

The developed computer aided instruction softwares lack the audio function units. But most of the students appear good reaction and acceptable status. The audio function attracts them understanding their living environment.

5. REFERENCE

[1] Shun Go, Gou-Shin, The Autism Message: Discovering Autism, Understanding Autism, Helping Them, Taipei, Lyan-Jing Book Ltd, 2000.

[2] Chen, Lyan-Shing and Shyu-Fen Hung, Digital School's Teaching and Learning-The Fundamentals and Applications of Network Teaching, Koachung, ShrRu Management Consultant, 2003.

[3] Chang, Jya-Yi and Hung De-Sheng, The Discussion of Future Education on e-learning, Information and Education Magazine, pp.93-103, 2002.

[4] Hung, R.H., The e-learning's Development Strategies on Campus, Teachers, vol.179, pp.26-28, 2004.

[5] Lin, Dung-Chin, Knowledge Management, Taipei, Jr-Sheng Culture, 2004.

[6] Hung, Ch-You, An Introduction on IT Knowledge Management, Taipei, Wen-Que, 2003.

[7] Lin, Hai-Ching, Knowledge Management and Education Development, Taipei, Yuan-Jau, 2002.

[8] The Nation Education Data Museum, Education Dictionary, Taipei, Wu-Nan Book Ltd, 2003.

[9] Wu, J.M. and J.H. Chao, The First Network Teaching Area, Information and Education, pp.35-43, 2003.

[10] The Special Education Association of ROC, Special Education Course, Taipei, Psychology, 1993.

[11] Tsao, Chun-Joan, The Autism Children and Education Cure, Taipei, Psychology, 1994.

[12] Mark, Rosinburger, e-learning: The Final Strategy on Increasing Personal Competition, Strengthening Business Advantages, Taipei, McGrill, 2001.

[13] Clark, R., & Mayer, R., e-Learning and the science of instruction：Proven guidelines for consumers and designers of multimedia learning. San Francisco：Jossey-Bass/Pfeiffer, 2003.

[14] Jyang, N.F., The Discussion of the Teaching Environment Establishment Centered by the Learners, Journal of Koachung N. Univ., vol.13, pp.3-15, 2002.

[15] The Special Education Team of the MOE. http://www.edu.tw/EDU_WEB/Web/SPECIAL

[16] The National Special Education www.spc.ntnu.edu.tw

[17] The Special Education Association of ROC http://searoc.aide.gov.tw/

[18] Wang, Tyan-Mao, The Manuals of Special Education Professional Service, Taipei: The Special Education Team of Ministry of Education., 2001.

[19] The Foundation of Autism of ROC http://www.fact.org.tw/

[20] Mao, Lyan-Wen, Special Children Education Method, Taipei, Psychology, 1999.

[21] Liu, Mei-Ron, The Truth of Autism, Taipei, Psychology, 2000.

[22] Yang, Gwei-Fen, Ts-Ai Hung and Mei-Hwei Wang, The Skill Training of Autism Children's Social Emotion, Taipei, Psychology, 2003.

[23] Wu, Hwa-Gwo, Special Education Courses and Teaching, Taipei, 2004.

A SHARABLE TEST CONTENT DEVELOPMENT AND ANALYSIS MECHANISM BASED ON SCORM

Lun-Ping Hung, Hsuan-Che Yang**, Wen-Chih Chang**, Hui-Fen Chiang**

*Dep. Of Information Management, Northern Taiwan Institute of Science and Technology
** Dep. of Computer Science and Information Engineering, Tamkang University
Taiwan, R.O.C.
E-mail: robin@ntist.edu.tw

ABSTRACT

In the environment of distance learning, most of current systems available are unable to share the experience of developing material content and can not provide subsequent learning result analysis. In order to overcome these shortcomings, we developed a sharable test content development and analysis mechanism (STCDAM) based on SCORM (Shareable Content Object Reference Model) and item response theory (IRT). Our purpose is to construct an integrated distance learning systematic assessment mechanism in which test content is shareable and the quality of a test can be improved. STCDAM incorporates advanced multimedia tools for content builder to construct test content easily based on the requirement of SCORM and evaluates online learners' overall learning performance for teachers to adjust the test content to better reflect students' learning ability.

Key words: distance learning, systematic assessment mechanism, SCORM, item response theory

1. INTRODUCTION

Due to the rapid development of internet infrastructure, distance learning mechanism built upon the internet plays an important role in a diversified learning environment. Therefore, learning is not constrained by time and place. However, how to capture online students' learning performance is widely discussed by researchers and system developers. The development of intelligent analytical mechanism which uses computer technology to analyze student's received knowledge will influence the development of distance learning in the future. Therefore, we concentrate on the construction of an intelligent analytical mechanism to evaluate students' learning performance and developed an online evaluation system to refine test content to better reflect students' learning ability. This paper is organized as follows. In section 2, we briefly describe the basic structure of our proposed sharable evaluation system STCDAM which includes the sharable test content creator and the online evaluation system. Conclusions are presented in section 4.

2. RELATED WORK

Regarding the build-in test content database system, it is designed to meet various requirements by different professions. For example, for math test, test content database system utilizes JAVA language to establish the test content [1, 2]. These test content can not be reused for the next text which may have few difference from the previous test. We developed a user friendly test content creator for teachers to build up the test content easily and quickly. Moreover, there are many researchers devote themselves to the study of evaluation system. Algorithms are proposed to measure students' learning performance [3, 4]. The most popular method is to evaluate students' learning performance based on students' grades received on tests. Our study is inspired by the item response theory (IRT) proposed by Lord [5] and define the level of discrimination and difficulty of a test content in order to adjust and refine the test quality.

Setting up a complete test content database system is not a easy task. If the work of one teacher can be shared among all teachers, they can absorb the spirit of others work. Moreover, if one type of test content systems can be transferred among different types of systems, the work can be duplicated. Therefore, a lot of standards are established all over the world to solve the problems of teaching material transferring among different systems, circulating among teachers and reorganizing between different tests. Such standards include, SCORM (Sharable Content Object Reference Model) [6] proposed by the United States government, QTI LIP content Packaging proposed by IMS [7], and AGRs and CMI proposed by AICC (Aviation Industry CBT Committee) [8]. SCORM is the most widely accepted standard. Therefore, in this paper, we use SCORM as a standard to establish test content.

Figure 1: An example of packing test content based on SCORM

3. THE SHARABLE TEST CONTENT DEVELOPMENT AND ANALYSIS MECHANISM (STCDAM)

The STCDAM system includes two important subsystems: the sharable test content creator and the online evaluation analyzer. The first system helps the teacher to develop the test content and the second system can help the teacher to capture students' learning performance in order to refine the quality of the test content.

3.1. Sharable test content creator based on SCORM

The shareable test content creator can be used by the instructor or the test content database provider to construct standardized test content database and to edit the content of a test. This subsystem meets the standard of SCORM which allows teaching material can be shared among different users. SCORM was proposed by the U.S. federal government in 1997 and contains the subject of development, packing, and circulation of teaching material. The object of establishing SCORM is to create a reusable, accessible, interoperable, and durable teaching material. In this section, an example is given to explain the process of packing a sharable test content which meets SCORM standard.

There are two steps to complete of the process of constructing the shared test content based on SCORM. The first step is to set up a imsmanifest.xml file which contains several parts. The first part of the file (metadata) declares the supporting information and the following part (resources) indicates the resource of the teaching material. The third part (organization default) describes information regarding this organization. Finally, the last part (manifest identifier) records all of the related files, including files, folders, java script files, and xml files.

Table 1: Metadata for test content searching

Metadata Name	Exam Name	Problem Name	Subject	Level	Date	Keyword	Problem Type
Attribute Value	User Defined	User Defined	Ex. Math	{1,2,3,4,5}	Ex.2003/4/2	User Defined	{True-False, Multiple Choice, Fill-in Blank}

Table 2: Metadata for test content annotation

Metadata Name	Language	Charset	Subject	Level	Computer Scored	Hint Permitted	Feedback Permitted
Description	International language set	Character Set	Educational subject set	Problem difficulty level, 1:very easy	Scored by computer or not	Hint permitted or not	Feedback permitted or not
Attribute Value	Default: English	Default: Big5	Default: Math	{1,2,3,4,5}	{Yes/No}	{Yes/No}	{Yes/No}
Example	Chinese	Big5	Math	3	No	Yes	No

The second step is to pack all of the necessary files indicated in imsmanifest.xml file together with the imsmanifest.xml file to become one PIF file (Package Interchange Format). Figure 1 is an example of packing test content based on SCORM. An important step in setting up the SCORM based test content is to establish metadata for test content searching and for test content itself. Table 1 explains the detail of establishing metadata for test content searching and table 2 illustrates an example of creating metadata for test content annotation.

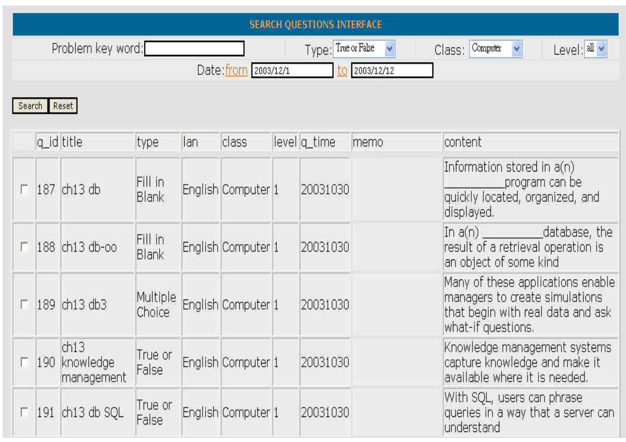


Figure 2: Question searching interface

In spite of packing the shareable test content and creating metadata for test content, the test content creator also provide a user friendly editing tool for instructor to prepare a test quickly and easily. As shown in figure 2, by entering few questions, such as problem key work, question type, instructor can select desired questions pulled out from the question database.

3.2. On-Line evaluation analyzer

In this section, the mechanism of improving the quality of a test based on students' test result in each question is demonstrated. Each question in a test is reviewed by the online evaluation analyzer and the analyzer suggests the appropriateness of each question. With the help of suggestions proposed by the analyzer, instructor can decide whether to leave the question in a test or to eliminate the question from a test.

First, each question is identified with five statistic characters: No, PH, PL, D(=PH-PL), and P(=(PH+PL)/2). No denotes the number of the question. PH indicates the percentage of students whose score is in the highest 25 percent answering this question correctly. On the contrary, PL indicates the percentage of students whose score is in the lowest 25 percent answering this question correctly. D represents this question's level of discrimination. P represents this questions' level of difficulty. Example of denoting problem attribute for a multiple choice question is illustrated in table 3.

Table 3: Problem Attribute

	Option A	Option B	Option C	Option D	Option E
High Score Group	HA	HB	HC	HD	HE
Low Score Group	LA	LB	LC	LD	LE

For a multiple choice question with five options, the number of students who belong to high score group and low score group should be calculated for each option. HA denotes the number of students who are in high score group choose option A and LA means the number of students who are in low score group choose option A, and so on. These numbers are used for measuring each option in a multiple choice question. There are four rules to analyze each question in order to distinguish the justice of each option in a question.

RULE 1: If (LA|LB|LC|LD|LE)=0, then this option shows a low attractiveness.

RULE 2: Let N={A, B, C, D, E}

If N is the correct answer among five options and the number of students in high score group who pick N is less than the number of students in low score group who pick N, then this option is problematic.

If N is the wrong answer among five options and the number of students in high score group who choose N is more than the number of students in low score group who choose N, then this option is problematic. This rule give the instructor a clear view to see if there is any option which is not clearly stated and should be revised.

RULE 3: Let LM=MAX(LA,LB,LC ,LD ,LE),
Lm=min(LA ,LB ,LC ,LD ,LE) and
LS=LA+LB+LC+LD+LE

If |LM-Lm | LS*20%, meaning students who receive lower score are not familiar with this question because students are equally chosen among options. On the other hand, if the difference between LM and Lm is larger than 20 percent of the total number of students in low score group, this means most of low score students choose certain option.

RULE 4: Let HM=MAX(HA ,HB ,HC ,HD ,HE),
Hm=min(HA ,HB ,HC ,HD ,HE) and
HS=HA+HB+HC+HD+HE

If |HM-Hm | ≦ HS*20% AND |LM-Lm | ≦ LS*20%, meaning no matter what type of students, either students receiving higher score or students receiving lower scores,

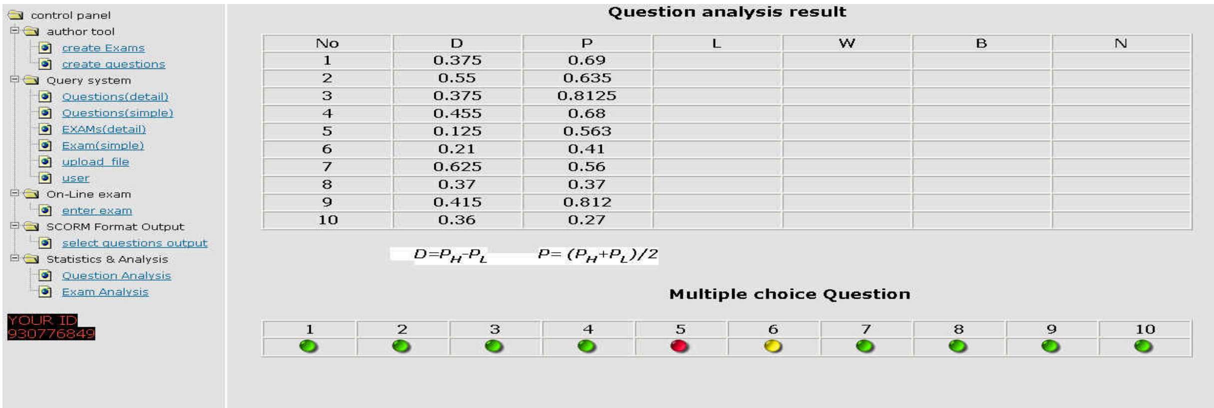


Figure 3: question evaluation interface

are not familiar with this option. The instructor should seriously consider to revise the content of this option.

After analyzing each question, we will able to provide suggestions for each question based on previous analysis. Figure 3 shows an example of analysis result for ten multiple choice questions. According to table 4, instructor knows the status of each question and should decide to adjust or to eliminate inappropriate questions. Questions which receive the value between 0.2 and 0.3 should be adjusted and assigned with a yellow button. If the value of D is less than 0.2, this question is considered inappropriate and should be removed from the test and then assign a red button to this question. As shown in figure 3, question number 5 showing a red button is absolutely inappropriate and question number 6 showing a yellow button should be adjusted for students to better understand these questions.

Table 4: Status of each question

Status	Light	D
Good	Green	> 0.3
Amendable	Yellow	0.2-0.29
Inappropriate	Red	0.19 <

4. CONCLUSIONS

With the rapid development of internet, information available online is abundant. How to quickly and correctly absorb the spirit of others' work becomes the key to success. A lot of researchers make efforts to establish information standardized in order to search information quickly and correctly. The development of SCORM meets their goal. SCORM allows information to be sharable and reusable among different systems and users. Our proposed STCDAM is base on SCORM version 1.3 and incorporates two subsystems: sharable test content creator and online evaluation analyzer. The first subsystem enable instructor to prepare a SCORM based test content. The second subsystem provides an evaluation mechanism for instructor to capture students' learning performance in order to refine the quality of test content. STCDAM provides a method to reassure the justice of test content.

5. REFERENCES

[1] Chiou-Hwa Lu, Han-Pang Huang, "Java-based distance learning environment for electronic instruments", , IEEE Transactions on Education, Volume 46, Issue 1,., Page 88 - 94, Feb 2003.

[2] McGough, J., Mortensen, J.; Johnson, J., & Fadali, S., "A web-based testing system with dynamic question generation"; Frontiers in Education Conference, 31st Annual, vol. 3, Page S3C 23-28, Oct. 10-13, 2001.

[3] Timothy K. Shih and Nigel H. Lin, An Intelligent E-Learning System with Authoring and Assessment Mechanism, in Proceedings of the 2003 International Conference on Advanced Information Networking and Applications (AINA'03), Xi'an, China, March 27 – 29, 2003.

[4] Timothy K. Shih, Shi-Kuo Chang, Ching-Sheng Wang, Jianhua Ma, and Runhe Huang, "An Adaptive Tutoring Machine Based on Web Learning Assessment," in Proceedings of the IEEE International Conference on Multimedia and Expo 2000, New York City, USA, July 31 - August 2, 2000.

[5]Lord, F. M. (1980). "Applications of Item Response Theory to Practical Testing Problems", Hillsdale, NJ: Lawrence Erlbaum Associates.

[6]SCORM (Sharable Content Object Reference Model) http://www.adlnet.org/Scorm/scorm.cfm

[7]IMS (Instructional Management System)

[8]AICC (Aviation Industry CBT Committee) http://www.aicc.org

Investigating the Transactions of Distance Learning System by Using Petri Nets Graph

Ko-Shung Chen*, Chiao Yu*, and Chon-Gi Chen+

*Department of Information Management, St. John's & St. Mary's Institute of Technology
+Department of Mechanical Engineering, St. John's & St. Mary's Institute of Technology
Tamsui, Taipei, 25135 Taiwan
{kschen, jo, cgchen}@mail.sjsmit.edu.tw

Abstract

In this paper, we exploited the discrete event driven model, Petri net model, to investigate the handshaking communication between server and clients of the distance learning system (DLS). In order to illustrate the processes of transactions between server and clients, we examine the process steps to access the desired data for the clients of DLS. The working model is modeled by Petri net simulation tool, Petri.NET simulation 2.0, to simulate our proposed DLS.

***Keyword**: Communication handshaking, Petri net (PN), non-uniform memory access (NUMA), PACE starter.*

1. Introduction

If a computer owns a privately local memory, we call it non-uniform memory access (NUMA) architecture. Otherwise, it is called uniform memory access (UMA) architecture as shown in Figure 1. For the NUMA architecture, every processor can access each hardware resource; however the access cost is not uniform [6]. A distributed system must provide various mechanisms for processor synchronization and communication, to deal properly with deadlock problem and to cope with all kinds of complicated failures that are not encountered in a centralized system. The communication patterns are determined to algorithms and protocols are used as well as by the architectural support provided. Frequently encountered communications patterns include permutations (also called unicast), broadcast (one-to-all), multicast (one-to-many), and conference (many-to-many) communication. Those communication demands may limit to the granularity or parallelism [3, 4].

The communication latency is the one of benchmarks of multiple processor (MP) systems and the distance learning system (DLS) as well. The structure of distance learning system likes that of MP. The communication latency is defined to *the total time to send message transmutation through the inter-client (processor) communication*. In fact, the communication latency imposes a limiting factor on the scalability of the communication mechanism size. For example, communication latency increases with respect to proposed DLS capacity. Thus, proposed DLS cannot be increased indefinitely without exceeding the tolerance level of the access latency [1, 4].

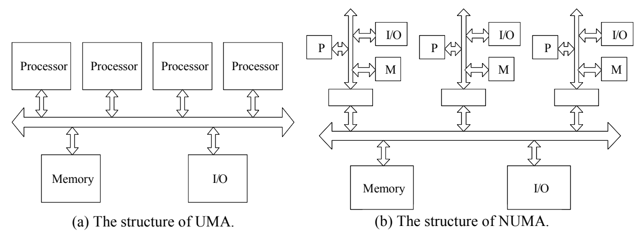


Figure 1: The architectures of UMA and NUMA.

Moreover, the communication mechanism should be provided for accessing message pertaining to other processors. The communication issue thus involves the reduction of latency or complexity, the prevention of deadlock, minimizing blocking in communication patterns, and the trade-off between parallelism and communication overhead. In this paper, we represent a communication protocol to minimize the communication latency for applying to distance learning and illustrate the communication model by using the Petri net model [5, 7].

2. The Protocol of Communication Handshaking

A communication network can be represented by a communication graph in which vertices correspond to the switching elements of the parallel computer and edges represent communication links. According to the constructed topology, interconnection networks can be classified as static or dynamic networks. In static networks, the connection of switching units is fixed and typically realized as direct or point-to-point connection. These networks are also called direct networks. In dynamic networks, communication links can be reconfigured through a round of settings by the active switching units of the system [1].

In this paper, we proposed a novel communication handshaking scheme for the inter-processor communication network, which compounds the multicast architecture, queue model, and interrupt controller for DLS. Simulation results reveal that this scheme can markedly reduce the communication latency overhead.

A distributed system is a collection of loosely coupled processors interconnected by a communication network.

This work is supported by the National Science Council, R.O.C., under contract no. NSC 93-2213-E-129-010

From the point of view of a specific processor in a distributed system, the rest of the processors with respective resources are remote, whereas its own resources are local. The designer of a communication network has to address four basic issues:

1) Naming and name resolution: how do to processes locate each other to communication?
2) Routing strategies: how are messages sent through the network?
3) Collection strategies: how does server send a sequence of messages?
4) Contention: due to network resources are share, how do we resolve the conflict of network access?

Communication control can take place only if the correct processor access rights are granted. Each processor actually has a message queue, which holds sent messages. When the message queue is filled with, the initiating processor, also called sender, can do one of the following three things [4]:

1) Sender, also called server, can block the request until it is resent successfully;
2) Sender can ask the communication unit to hold the request message temporarily;
3) Sender can be returned an error message from client when the client doesn't correctly receive the desired data.

For any client of the DLS may retrieve data from server. The requesting processor must first send a request message, in which contains the address of the data, to the server. The request message is transferred to the server via the communication network. If the server in busy mode, the request message is buffered in the request message queue until the server reverts to free mode. When the request arrived, the server must send a reply message with the desired data to the requesting processor through the data queue of the communication unit. The proposed handshaking protocol sketched in signaling diagram is shown in Figure 2. In addition, the algorithm of communication process for the proposed DLS is illustrated in Figure 3.

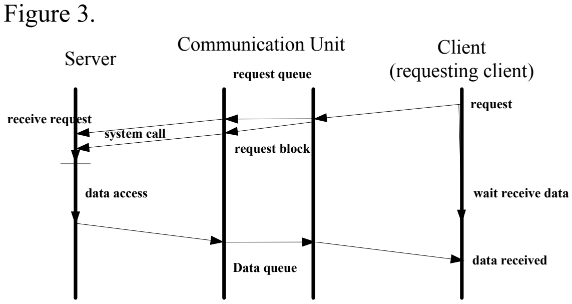


Figure 2: The protocol of communication handshaking of the proposed DLS.

The proposed communication mechanism is constructed of request buffer and data buffer for buffering the request for requesting processor and the required data from the server, request message register for registering the request when the server is block or busy, and interrupt controller for initializing the server to access the required data from its local memory. The request and data buffer is a FIFO (first in first out) structure to hold the request message and the accessed data for the requesting processor. In practice, the buffers can only be finite length and a critical point could be reached when the routines were hung up because all the available buffer space has been exhausted. This communication mechanism allows the transmission of message and data between the executable units of processors.

```
/* The algorithm of communication */
Repeat:
:
requesting processor has an request item in next-tp of
the request buffer of communication unit;
:
/* sending the request message to the desired
processor */
send (desired processor, next-tp);
until false;
the desired processor process is defined as the
required for the requesting processor;

/*receiving the required data from the data buffer for
the requesting processor */

receive (requesting processor, next-d);
:
desired processor send the data item to the next-d of
data buffer;
:
until false;
```

Figure 3: The algorithm of this proposed DLS.

For a receive routine, the required data has to have been received if it is sent to the data buffer of the communication unit. If the receive routine of the requesting processor is reached before the send routine of the server, the data buffer of the communication will be empty and receive routine must wait for the required data. However for the send routine of the server, once the local actions have been completed and the required data is safely accessed on its way, the process of the requesting processor can continue with subsequent work and instruction. In this way, we use such receive and send routines interaction between requesting client and server that can notably reduce the overhead of communication time. Extensive simulations will exhibit the noticeable effect.

For examining the basic message parameters associated with sender and receiver system calls, some message-passing primitives are specified as follows.

- To create and to delete the communication connect buffer, request buffer and data buffer, via the *request* and *data_receiver* routines;
- To send the request and to receive the required data among processors;
- To transfer status information; and

- To attach or detach remote processors, such as requesting and servers.

The send and receive primitives of the communication unit are implemented by the sending and receiving processes, respectively. Therefore, the buffer field of send message specifies the memory location of the data to be retrieved from the server. On the other hand, the buffer field in receive message specifies where the arriving data will be stored before to be read by the requesting processor.

3. Petri Net Model for Inter-Processor Communication

In this session, we use a communication unit to manage the request message and the required data transmission between interconnected processors. Figure 4 illustrates the communication topology of the proposed DLS. In order to illustrate the model of communication handshaking of DLS, we also modeled the proposed DLS by using a discrete event model tool, Petri net [5, 7]. Figure 5 presents how to use the Petri nets for modeling the proposed DLS. Referring to Figure 5, when a data is desired to the executing processor; it is accessed from either the privative memory of the executing client or server. If the required data is accessed from the server, the communication handshaking starts the place of *request data from other* for the requesting processor, then initialize to the transition of *request queue*, and feed to the place of *request queue.* If the server is busy, the request is temperately queue in the place of request register; else the server is initiated the *request interrupt* transition to access the required data via the *system call* transition.

The required data are transferred from server to the requesting processor through the *received data* transition and the *data queue* place. If the required data is failure to access from the server, the respondent signal still blocking in the *ready to receive data* place until the required data is completed to the server.

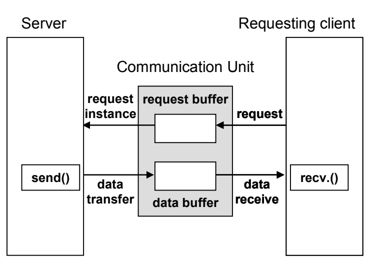


Figure 4: The topology of DSL.

4. Simulation Results

To perform simulation, we adopted a simulation tool – *Petri.NET simulation 2.0* [2] to generate the realization of Petri net model. The simulation block diagram is demonstrated as Figure 6. In Fig. 6, we simplified the architecture of the DLS with Petri net model where IA indicates the server, IB, OA, and OB stand for the clients of DLS, and S1 presents the communication unit to handle the handshaking and communication of the command and data between the server and clients, respectively. This simulation system is in a timed-event driven manner, the total simulation time units and the period of token issue are set to be 20000 and 10, respectively. We also assume that each client has the same priority to issue its request to the server, IA place.

For the simulation model of Figure 6, in which have four types components, there are place node, transaction node, arc, and token. The place node, represented by a circle, indicates the processed status to contain the related conditions and information. The transaction node, represented by a line or box, indicates the occurred event. Arcs connect place nodes to transaction nodes and transaction nodes to place nodes. A token is indicated by a dark dot positioned in the appropriate place. The state transition mechanism in Petri net is provided by moving tokens through the net and hence changing the state of the Petri net.

Figure 7 shows the simulation results of the proposed DLS. The initial token number of IA place node is 10 that is the maximum capacity of the server to support 10 users, clients, at the same time. For IB place node, the token number indicates it has the maximum receiving capacity for same time. For OA and OB place nodes, the capacity is one that is them only issue one request and receive one material from the server, IA place node, via the net and communication unit, SI node. Refer to Figure 7, we find the OA and OB place node have higher access frequency of tokens than to the place node IB. Though, the OA and OB place nodes have long distance than IB place node to the server, respectively. By the proposed communication unit and scheme, we find the communication latency is evidently to be improved of OA and OB place node yet.

5. Conclusions

This paper described a novel construction method of proposed DLS for multiprocessor communication in distributed system. We first illustrate the proposed DLS model with discrete event model, Petri net model. Furthermore, to analyze the communication latency to verify our proposed DLS with Petri net simulation tool, *Petri.NET simulation 2.0*. Simulation results reveal that this scheme can markedly reduce the communication latency.

References

[1] Kai Hwang, "Advanced Computer Architecture Parallelism Scalability Programmability," *McGraw-Hill, Inc.* International Editions 1997.

[2] IBE Engineering Office, "Petri.NET simulation 2.0," *IBE Engineering Office*, 1999.

[3] B. J. Kim, H. S. Kim, and B. Ahn, "Design and Implementation of IPC Network in ATM Switching System," *The Proceedings of 5th International Computer and Communication Conference*, pp. 148-152, 2001.

[4] R. Kobayashi, M. Iwata, Y. Ogawa, H. Ando, and T. Shimada, "An On-Chip Multiprocessor Architecture with a Non-Blocking Synchronization Mechanism," *The proceedings of 25th EUROMICRO Conference*, Vol. 1, pp. 432-440, 1999.

[5] T. Murata, "Petri Nets: Properties, Analysis and Applications," *Proceedings of the IEEE,* Vol. 77, No. 4, pp. 541-580, Aril 1989.

[6] C. A. Petri, "Communication with Automate," *New York: Griffiss Air Force Base*, technical report RADC-TR-65-377, Vol. 1, Suppl. 1, 1996.

[7] R. R. Razouk, "The Use of Petri Nets for Modeling Pipelined Processors," *On Proceeding of the 5th ACM/IEEE Design Automation Conference*, pp. 548-553, 1988.

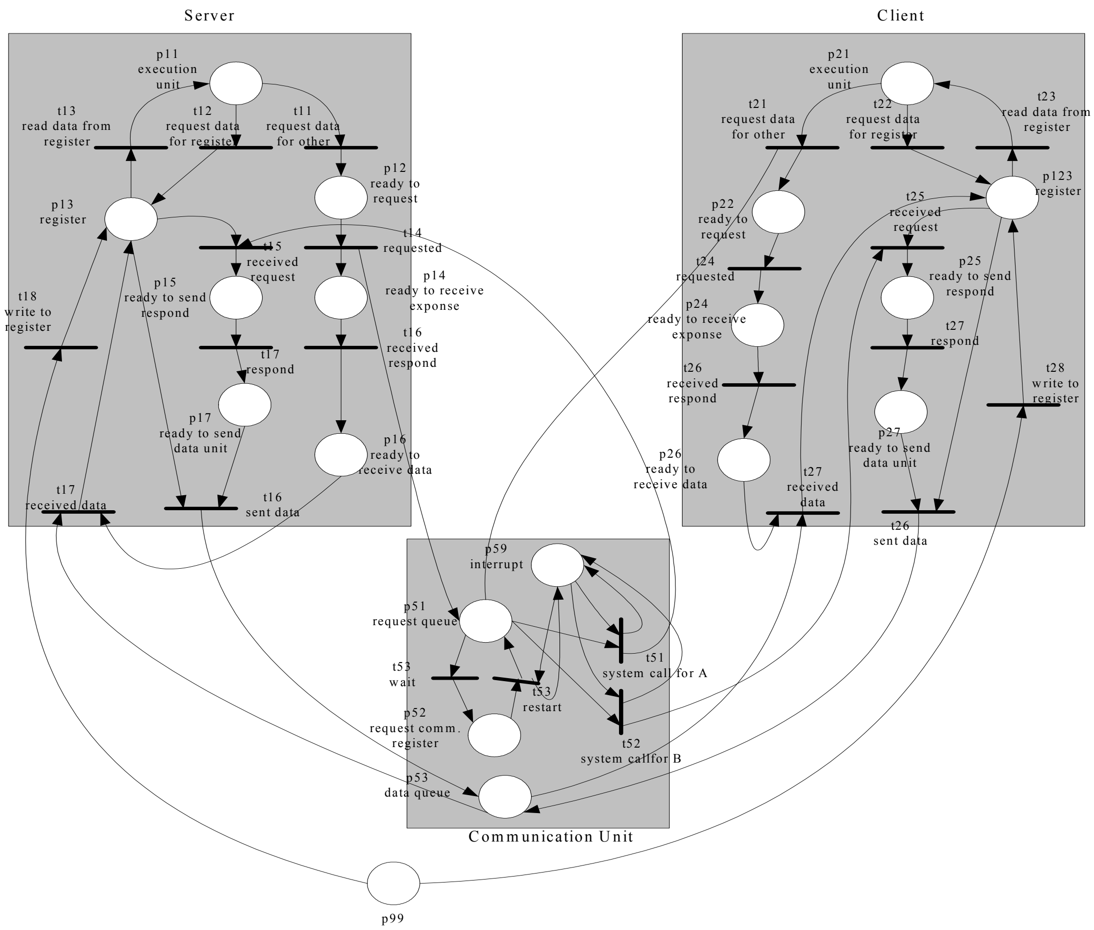


Figure 5: The Petri net model for DLS.

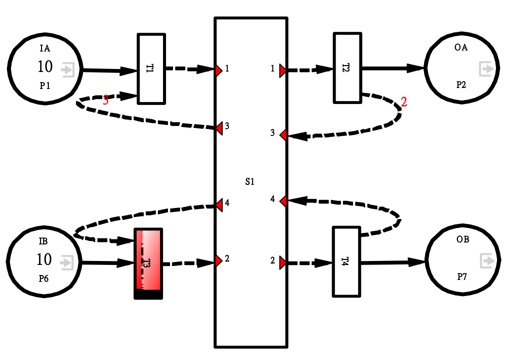


Figure 6: The simulation model with Petri net model for DLS.

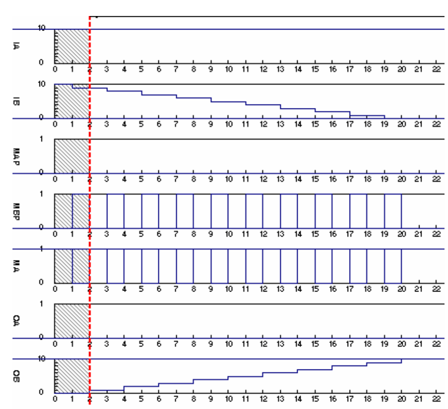


Figure 7: The simulation results of the proposed DLS.

Exploiting Discrete Computer System to Architect for Distance Learning System

Jih-Fu Tu[1] Chih-Yung Chen[2] Chun-liang Hsu[3]

[1]Department of Electronic Engineering
[2]Department of Information Management
[3]Department of Electrical Engineering
St. John's & St. Mary's Institute of Technology
{tu, yung}@mail.sjsmit.edu.tw

Abstract

For improving the performance of distance learning system (DLS), we will propose a communication unit to perform high-speed data transmission between processors of the discrete computer system (DCS). It is implemented by hybrid technique, in which is combination of the schemes of interrupt, queue-network, and non-blocking. In order to delicately describe the state transform, we also exploit the discrete event driven model, Petri net model, to illustrate the communication handshaking of the inter-processor of DCS. Final, we evaluate the communication delay time when the desired data of DLS is transmitted among processors of DCS with different kinds of size of the request-queue and data-queue of communication unit. We also compare the performance of this proposed communication unit to that crossbar-based network topology structure.

Keywords: Distance learning system (DLS), communication network, discrete topology, queue network, and discrete computer system (DCS).

1. Introduction

A discrete computer system is a collection of processors that may or may not share the memory or a clock. If it owns privately local memory, we call it non-uniform memory access architecture (NUMA). Otherwise, it is called uniform memory access architecture (UMA), in which all processors have equal access time to all memory words [2,3,5]. The NUMA architecture is a shared-memory system. Each processor can access each hardware resource, but the access cost is not uniform.

The quality of the discrete computer system (DCS), also called tightly coupled system, has a decisive impact on the speed, size and cost of the whole architecture. In discrete computer system, any processors must be able to access any memory location, even if it physically belongs to another processor. Thus, the communication interconnection schemes are usually employed. The communication machine of discrete computer system allows the messages to be transmitted between execution units of different processor [1,5].

The communication interconnection topology can be further classified into two types: signal bus and multiple bus systems that is depended on the number of buses providing interconnection among the processors, called node, of the discrete computer system architectures.

The structure of a multiple bus discrete computer system is combination of multiple processor, bus, memory module, and one bus arbiter. The architecture of a 1-dimension multiple bus multiprocessors are shown in Figure 1, which is called a b-of-m arbiter in Mudge et al. [7].

In this paper, we propose a multiple bus for this proposed DCS, which is implemented by the schemes of interrupt, queue-network [8], and non-blocking [6], to improve the performance of this DCS and to reduce the communication latency time of the DLS, respectively.

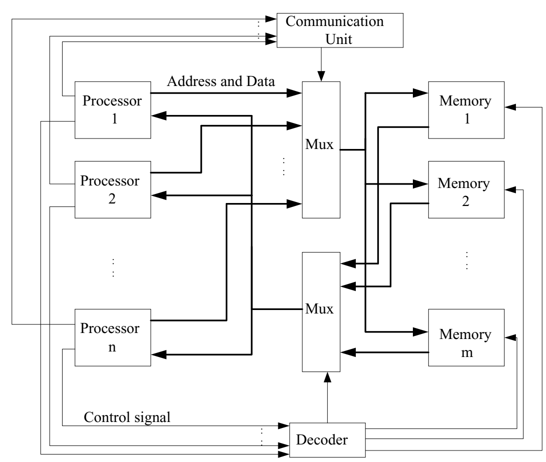


Fig. 1. The diagram of b-of-m structure.

The communication latency time is the important one of benchmarks of interconnecting processors of DCS [5,7]. The communication delay time is defined to *the total time of the request is issued to the desired data are received*. In fact, the communication latency imposes a limiting factor on the scalability of the communication mechanism size. For example, communication latency increases with respect to the capacitance of DCS. Thus, the number of processors of DCS cannot be increased indefinitely without exceeding the tolerance level of the access latency. For a DCS, the desired data of each processor, also called client or node, may be accessed through two sources, the local memory (called P-to-M or P2M) or other processor (called P-to-P or P2P) [10,15,16].

Also, the communication mechanism should be provided to access message pertaining to other processors. Thus, the problems of communication involve the reduction of latency or complexity, the prevention of deadlock, minimizing blocking in communication patterns, and the trade-off between parallelism and communication overhead. In this paper, we not only propose a communication protocol to minimize the communication latency, but also provide the maximum communication throughput via the proposed DLS architecture.

A communication network can be represented by a communication graph. The topology of the communication graph is an important property, which significantly influences latency on multiprocessors. Thus, there are two-communication models, message-passing model and share-memory model, are discussed and analyzed. For the message-passing model, the information is exchange through an inter-process communication facility provided by the communication routine [11,13]. In the share-memory model, processes use map memory scheme to gain access to regions of memory owned by other processors.

The recent of this paper is organized as follows. Section 2 surveys the previous works of communication for the DCS or multiple processors. Section 3 illustrates the Petri net for modeling the communication machine. The communication latency and performance of DCS is analyzed in Section 4. Finally, we remark the conclusions in Section 5.

2. Previous Works

A communication network can be represented by a communication graph in which vertices correspond to the active/positive process elements, i.e. processor/memory, of the parallel computer and edges represent communication links, i.e. bus. According to their topology interconnection network can be classified as static and dynamic networks. In static networks the connection of switching units its fixed and typically realized as direct or point-to-point connection. These networks are also called direct networks. In dynamic networks, communication links can be reconfigured by a setting the active process units of the system. For multi-computers are typically based on static networks, the dynamic networks are mainly employed in multiprocessor [14].

In the previous studies, the communication system between processors of multiprocessor architectures is accomplished by the following topologies network, such as crossbar, ATM, switch etc. Park [10] and Kim [5] exploited the ATM switch to construct the DCS. The latency overhead will be added when the request or data transfer from the source to destination processor. In 1999's, Kobayashi [6] proposed ring-bus scheme to connection the inter-processor of DCS, in which, the author added external a synchronization table and send controller to each processor element and it is a point-to-point architecture.

In this paper, we will propose a high-speed communication unit and bus arbitration circuit for arbitrating the bus competition among processors, and bus decoder for deciding the bus to be used by which one processor of DCS for the clients of DLS.

The total overhead of communication time, called communication latency, is defined that a message is transmitted between inter-processor, which can be expressed as follows:

$$\textit{Communication latency}\ (T_{latency}) = T_r + T_{rq} + T_{sd} + T_{dq} + T_s$$

where:

- The receiving overhead (T_r) is defined to the time since the requesting processor issues the request message in the communication unit until it receives the desired data.
- The send overhead (T_s) is defined to the duration from the required processor receives the request message from the communication unit and it responds the acknowledge signal to the communication unit.
- The send delay time (T_{sd}) is defined to the duration from the requested processor issues the acknowledge signal to the desired data is sent.
- The time of stall of request queue (T_{rq}) is defined to the time from the request signal is delivered to communication unit to the request signal is received by the requested processor.
- The time of stall of data queue (T_{dq}) is defined to the stalled time of the desired data resided in the data queue.

The sending and receiving process implements the sending and receiving primitives of the communication unit, respectively, therefore, the buffer characteristic in send specifies the memory location of the data to be retrieved from the requested processor. On the other hand, the buffer field in receive specifies where the arriving data will be stored before to be read by the requesting processor.

3. Petri Net Model the Structure of Inter-Processor Communication for DCS

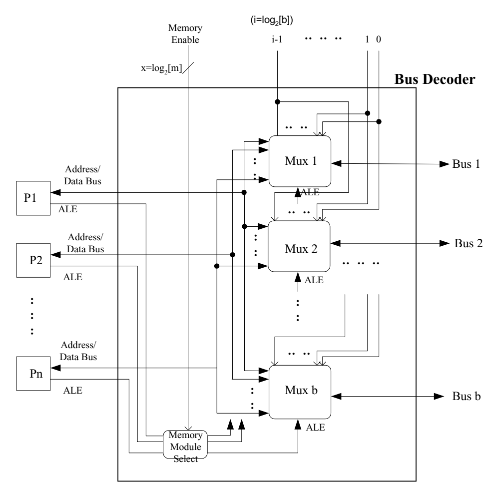


Fig. 2. The block diagram of the bus decoder of proposed DCS.

In this paper, we use a communication unit to manage the request message and the desired data transmission between interconnected processors. The structure of this proposed DCS, which is consisted of n numbers of processors, m numbers of memory-modules, b numbers of buses, and the communication unit, included the bus arbitration and the bus decoder. The structure of the bus decoder of this proposed DCS is shown in Figure 2, too.

The desired data of the requesting processor may access from one of two paths, other processor via remote produce call (RPC) or shared memory (memory module) via direct-access memory (DMA). In this paper, we only discussed with the data transmission between processor to processor.

In order to illustrate the communication model, we model this DCS using a discrete event model tool, Petri net [12,13, 14]. Referring to the state machine of p2p, we exploit Petri nets model to model the proposed DCS, which is shown in Figure 3, and illustrate the process as follows.

The architecture of the requesting or requested processor are similar to the DLX pipeline processor which has five pipeline stages, such as IF, ID, EXE, MEM, and WB [2]. The desired data of the execution unit is fed from two paths, the one is fetched from the place of *data memory* in the instruction decode pipeline stage then to the place of *register*. The other is severed from the place of *result store buffer* of the required processor via the communication unit in the MEM pipeline stage.

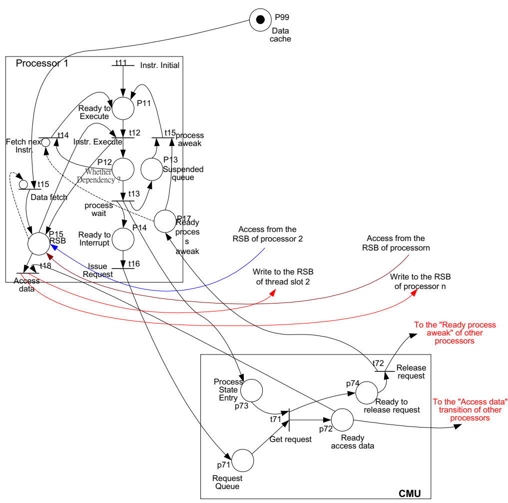


Fig 3 The Petri net model for DCS.

If the desired data is accessed from other processor (i.e. requested processor), the communication handshaking starts the place of request data from other for the requesting processor, then initialize to the transition of request queue, furthermore feed to the place of request queue. If the required processor is busy, the request is temperately queued in the place of request register; else the required processor is initiated the transition of request interrupt to access the desired data via the transition of system call.

The desired data are accessed to the requesting processor from the required processor through the transition of received data and the place of data queue. If the required processor does not generate the desired data, the respondent signal of the request is still blocked in the place of ready to receive data, until the requested processor completes the desired data.

4. Analyzing the Simulation Results

The communication latency is the one of performances of multiprocessors architecture. In this section, we analysis the communication latency of this proposed DCS for different number of PEs with different size of the request buffer and data buffer. The number of request- and data-buffer are various from 2 entries to 16 entries. We also compare the communication latency of the proposed communication unit to that of the crossbar based communication interconnection network.

This workload mechanism of the multiprocessor architecture in this paper is constructed as the UMA architecture, which specifications are defined as follows:

- the structure of PE is a five pipeline-stage processor,
- 512 Byte privative instruction and data cache,
- each entry of request- and data-buffer is 32-bit,
- request communication register has 16 entries, each entry has 32 bits,
  the latency time of request is 20 clocks, the latency time of data access is 15 clocks for basic mechanism
- one latency time is equal to 1 clock cycle, 1 clock cycle is equal to *1/the numbers of Hz of CPU.*

4.1 Comparison the Performance to the Crossbar Architecture

In this paper, we also compare the performance of this hybrid technology to that of the crossbar switch network architecture. The structure of the compared crossbar architecture is constructed to a 2*2 switchers. Figure 4 shows the simulation results of the proposed communication unit to that of the crossbar switching network scheme. We simulate those two different schemes with the same of data and instruction cache size, which are 2 K bytes and 4 K bytes, respectively. The proposed multiprocessor has 6-processor, 4-bus, 6 memory modules.

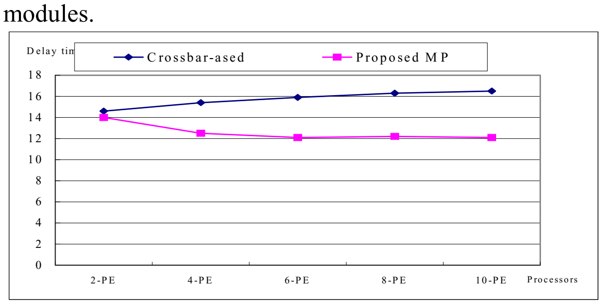


Fig. 4. The latency time of crossbar switching scheme and proposed hybrid scheme.

Referring to Figure 4, we find that the proposed DCS architecture has lower communication delay time than the crossbar switching network architecture. This is caused that the crossbar switching network architecture has higher overhead when data transfer via the crossbar switching. But this proposed communication architecture does not expend the switching delay time when the request or data is transmitted. In this proposed architecture, the desired data or commends are transmitted among processor is controlled by the bus arbiter and bus decoder, at the same time.

5. Conclusions

This paper describes a method of constructing the communication unit of the DCS and exploiting to the DLS. We not only illustrate the DCS model with discrete event model, Petri net model, but also analyze the communication *latency to cost* ratio under different numbers of request- and data-buffer, and the number of processor are changed of the proposed DCS. We also compare the performance of this proposed DCS to the crossbar switching network architecture. We find that the crossbar switching network architecture has higher switching overhead than this proposed communalization unit when the desired data and message are passing inter-processor. Thus, the proposed DCS can effectively improve the communication delay than the crossbar-switching network structure.

References

[1] Chih-Yung Chen, Jih-Fu Tu, Jenn Tang, Hsichien Yu, and Dong-Liang Lee, "Exploiting the Request-queue Methodology for Implementing Processor Communication," *Proceedings of the 2003 International Workshop on Distance Education Technology, DET'2003*, Step. 24-26, 2003, Miami, Florida, USA, pp. 814-817.

[2] H. B. Diab and H. S. Tabbars, " The Effect of Inter-PE Communication Network Configuration on the Performance or Array Computer, "*The Proceedings of 25th Annual Simulation Symposium*, 1992, pp. 164-171.

[3] J. L. Hennessy and David Patterson, "Computer Architecture A Quantitative Approach," Morgan Kaufmann Publishers, Inc. Second Edition 1996.

[4] Kai Hwang, "Advanced Computer Architecture Parallelism Scalability Programmability," *McGraw-Hill, Inc.* International Editions 1997.

[5] B. J. Kim, H. S. Kim, and B. Ahn," Design and Implementation of IPC Network in ATM Switching System," *The Proceedings of 5th International Computer and Communication Conference*, 2001, pp. 148-152.

[6] R. Kobayashi, M. Iwata, Y. Ogawa, H. Ando, and T. Shimada, "An On-Chip Multiprocessor Architecture with a Non-Blocking Synchronization Mechanism," *The proceedings of 25th EUROMICRO Conference*, Vol. 1, 1999, pp. 432-440.

[7] T. N. Mudge, J. P. Hayes, and D. C. Winsor, "Multiple Bus Architectures, IEEE Computer, Volume 20, Issue, 6, 1987, pp. 42-28.

[8] C. Mung, A. Sutivong, S. Boyd, "Efficient Nonlinear Optimizations of Queuing Systems," IEEE Global Telecommunications Conference, Vol. 3, 17-21 Nov. 2002, pp. 2425-2429.

[9] T. Murata, "Petri Nets: Properties, Analysis and Applications," *Proceedings of the IEEE,* Vol. 77, No. 4, Aril 1989, pp. 541-580.

[10]H.-S. Park, S.-J. Moon, M.-S. Park, B. Kwon, and K.-S. Song, "Design of Inter Processor Communication Controller using ATM Switch and Analysis of its Optimal Message Length Considering Retransmission," *The Proceedings of 4th International Workshop on Real-time Computing System and Applications*, 1997, pp. 18-25.

[11]P. G. Paulin, C. Pilkington, and E. Bensoudane, "Network Processing Challenges and an Experimental NPU [Network Processor Unit]," The Conference and Exhibition of Design, Automation and Test in Europe, 2003, pp. 64-69.

[12]C. A. Petri, "Communication with Automate", New York: Griffiss Air Force Base, Tech. Rep. RADC-TR-65-377, Vol. 1, Suppl. 1, 1996.

[13]R. R. Razouk, " The Use of Petri Nets for Modeling Pipelined Processors," *On Proceeding of the 5th ACM/IEEE Design Automation Conference*, 1988, pp. 548-553.

[14]D. Sima, T. Fountain, and P. Kacsuk, "Advanced Computer Architectures: A Design Space Approach," Addison-Wesley, 1995.

[15]J. F. Tu, "Using Coloured Petri net for Modeling a High-Performance of Processor Communication," *The Journal of SJSMIT*, Vol. 20, Taiwan, Oct. 2003, pp. 62-76.

[16]Jih-Fu Tu and Chih-Yung Chen, "An Effective Bus-Based Arbiter for Processors Communication," *The 18th International Conference on Advanced Information Networking and Applications (AINA 2004)*, March29-31, 2004, Fukuoka, Japan pp. 52-57

The Development of Digital Certificate on Distance Learning With Mobile Agent Technologies

Dong-liang Lee[1], Lawrence Y. Deng[2] and Ming-Chih Lu[3]
Dept. of [1]IM, [2]CSIE and [3]EE
St. John's Technology University
Tamsui, Taipei, R.O.C.
langlee@mail.sjsmit.edu.tw

Abstract

This paper proposed an application of distance learning which integrated with Citizen Digital Certificate (CDC) and Public Key Infrastructure (PKI), including the grade uploading, student information inquiring, and electronic documentation flow control. We integrated a tracking and persistent agent-based mobility management system in the case of distance learning. The main purpose of this system is addressed to achieve the universal access objective. We addressed to solve the problems in case of the data transferring in the network, such as eavesdropping, snooping, modification, masquerading and repudiation. And this would improve the efficiency of entire computerize procedure of electronic campus policy totally which in the following projects: Grade Processing System and Governmental Autographing System, etc. Especially, the Grade Uploading System was the first time to integrate with CDC. This experience was very important to integrate/generate the most secured application system in the Distance Learning.

Keywords: *Citizen Digital Certificate (CDC), Public Key Infrastructure (PKI), Distance Learning, Digital Certificate, Mobile Agent.*

1. Introduction

As the popular of the information technology and internet accessing, the distance learning was a very important issue in the world. As usual, the information broadcasting in the campus was based on traditional medium, like poster, leaf and campus news. At this point, there is a significant revolution on the limitation of the information transmission because of the real-time benefits on the network (include multimedia, communication, and high performance computer). This benefit also made user more disbelieve due to the open environment on the Internet; you could not get any protection on those digital documents which were transmitted on the network [1][2]. Therefore, the information security should be founded on a completed system that provided identification authentication, non-repudiation and exchangeable information. [3][4]

Besides, many marketing and technical terms under which agent supports in desktop, Internet, Intranet and so on. With the growing of network, likewise, the application-driven agents also provide many specialized facilities such as information retrieval agent, mobile agent, process automation agent, collaborative customization and database agent. The ever-increasing growth of mobile agent applications are encouraging research aimed at the wide spread communication infrastructure. In [5], the mobile agent paradigms and technologies were discussed. The relationships among paradigms and technologies for mobile agent are showed in Table 1.

Table 1. Relationships among paradigms and technologies

Technologies	Paradigms		
	Client-Server	Remote Evaluation	Mobile Agent
Message-based	Well-suited	Code as data Program interpretation	Code and stats as data Program state restoring Program interpretation
Weakly mobile	Code is a single instruction Creates unnecessary execution units	Well suited	States as data Program state restoring
Strongly mobile	Code is a single instruction Creates unnecessary execution units Move state back and forth	Manage migration Move state back and forth	Well suited

In this paper, we devoted our attention to the mobile agent issues especially in universal access objective. Since, mobile agents often work on heterogeneous network and operating system environment. Therefore, integrated logical interfaces to access physical structure via mobile agent application is become more and more important [6]. For flexibility, mobile agents can be accepted as a design paradigm like object-oriented programming or client/server computing [7]. So, we proposed a tracking and persistent agent-based mobility management system with security concerned in the case of distance learning to illustrate the entire processes.

The remaining parts of this paper were organized as follows: The related work was discussed in section 2. A mobile agent system architecture model describes server agent and client agent components are illustrated in section 3. The limitations and possible solutions are then discussed in section 4. And finally, we discuss our conclusion and extensions in section 5.

2. Related Work

2.1 Public Key Cryptography

The technology of Public Key Cryptography was proposed by Diffie and Hellman in 1976[8], and the first RSA Cryptosystem also developed by Rivest, Shamir and Adleman in 1977[9]. Each cryptosystem had its own characteristic in public key infrastructure individually. According to cryptology, the Key is a unit digits which created by a special way. The Public Key Infrastructure (PKI) system consist a pair of Keys --- the Public Key and the Private Key. Each pair of Keys was identical and unique, there were a corresponding mathematical relationship between the Public Key and the Private Key, and the generated procedure was irreversible. And the Private Key could not figure out from the Public Key practically <Security> [10]. The Public Key was open to public in PKI system, but the Private Key was saved in privacy. It was possible to decode with an encrypted Public Key which was pair to the specified Private Key previously <.Envelope>. At the same time, it also needed an encrypted Private Key to decode the coded data which pair to the specified Public Key formerly <Signature>. Therefore, we need to establish an acceptable Public Key on both side in the network, and this process should be verified by a certain procedure under a righteous Certificate Authority (CA) which had confidence in both two ends fairly. In the electronic identity authentication system, there was a common problem that the Certificate relationship was built between an end-user and particular computer system eventually. This was impossible to make this relationship expand to more end-users and more networking systems, and the public identity authentication system will not be come true at all. Therefore, PKI structure provided a certificate system for public identity authentication correctly, that is the correction of every user's authentication is accuracy and validity for sure.

2.2 Digital Signature

Please do not revise any of the current designations. Beside the style of an autographed document was different, the traditional stamp or personal signature would have the same content even the documents were not alike. This was the major difference between the traditional stamp that we used in Oriental and personal signature. Eventually, the Digital Signature was the calculated result of which generated through the Private Key that owned by the electronic document and dispatcher. And the Digital Signature was interrelated to the content of the electronic document. Therefore, there would have various Digital Signatures that came with numerous documents even they were generated by the same owner. Currently, the Digital Signature was based on the Open Public Key Cryptosystem of the Cryptography; it also named "Asymmetric Cryptosystem" as well. It was paired with Digital Signature and Digital Envelope basically. The dispatcher used his own Private Key to 'sign' with the messages by the Digital Signature that generated by the productive mechanism of signature. And the accepter of the Digital Signature would verify its validity by the dispatcher's Public Key that generated by the productive mechanism of signature similarly. This would make the contents of electronic document to give all consideration in confidential, integrity, authentication and non-repudiation [11]. The productive mechanism of signature was a method or a procedure which produced a Digital Signature by the dispatcher; it also treated as a mathematical algorithm. When the dispatcher signed the Digital Signature, there was a value need to input in the algorithm with his own Private Key and electronic document which would be signed. And the Digital Signature of this electronic document would be obtained after the execution of the algorithm. The productive mechanism of signature was also useful to accepter to verify the validity of the method or procedure in Digital Signature. When the accepter received the signed electronic document which delivered from the dispatcher, it was necessary to verify the validity for this Digital Signature through previous mechanism, including electronic document, Digital Signature and the Public Key of the dispatcher.

2.3 Mobile Agents

Mobile Agents is an agent encapsulates the program that a receiving server is to execute, the data comprising the program's arguments and state. Figure 1 shows a mobile agent scenario. Software agents that transport themselves form a client computer to various servers for remote execution.

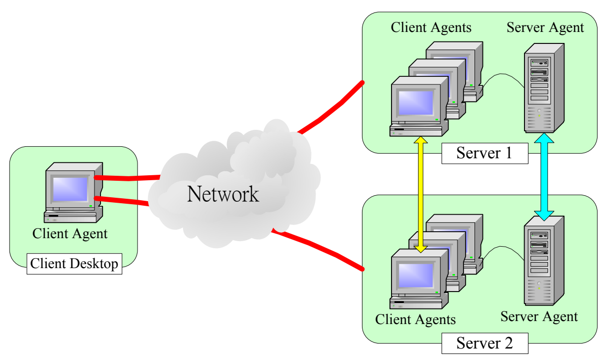


Figure 1. Mobile agent scenario

The concept of agent-based software engineering is discussed in a survey paper [12]. The author presents two important issues: agent architecture and agent communication language. Agent architecture, on the other hand, includes network infrastructure and software architecture that ensures agent computing. An open agent architecture for kiosk-based multimedia information service is proposed in [5]. Agent communication languages allow agents to share information and send message to each other.

2.4 Mobile Agent Programming Language

Mobile agent needs a programming language that lets users define the role of their agents as they travel across a network. The following languages can be used for specific niches as well.

KQML: A language developed through ARPA knowledge sharing effort aimed at developing sharable large-scale knowledge bases to enable agent-to-agent communication and knowledge. KQML is Suitable for development agent prototype for integration into existing application [13].

Java: Java is an interpreted, multithreaded, and secured platform-independent language suitable for agent application. The multithreading and built-in security features give Java an edge in the implementation of interactive e-commerce applications [14].

ActiveX/OLE: The programming technologies from Microsoft Corporation. Microsoft's answer to Java embracing both Java and Component Object Model (COM) based on Object Linking Embedded (OLE). [15]

Tcl and Safe-Tcl: This is a machine-independent scripting language that uses email to transport agent procedures as contents of mail messages [16].

Telscript: A language proposed by General Magic. Telescript is an object-oriented language and agent-based operating environment designed for mobile agent [17].

3. Mobile Agent System Architecture

Before In this paper, we proposed a model to characterize the mobile agent system architecture. As Figure 2 illustrated, the mobile agent virtual society was composed within three cells: Pico cell, Micro cell and Macro cell. The Pico cell represented the client mobile agent. The Micro cell constituted by at least one client mobile agents (Pico cells) and one server agent. Several Micro cells construct a Macro cell. The following article describes the architecture of the Micro cell (Server side) and the Pico cell (Client side).

3.1 Micro Cell

A micro cell is composed of agent profiles/database, server agent and web server.

Agent Profile and Database:

- User personal environment setting information: This information forms strings, including the items the user chose with their needs for the environment.
- User Log Files: The log file plays an important part while agent carried back to agent server. The agent server will parse the log files with different catalogs, such as the course participations, the shopping experience, and so on.
- User personal information: The user may change his/her own personal information via the agent, the modification of the member databases.
- User submitting results: There is some information, which users can submit via the agent architecture, such as the questionnaire system, the pop-up quiz system. Agents won't bring all of the submitted information, and some of that information will be sending back to the database with the functions provided in each subsystem.

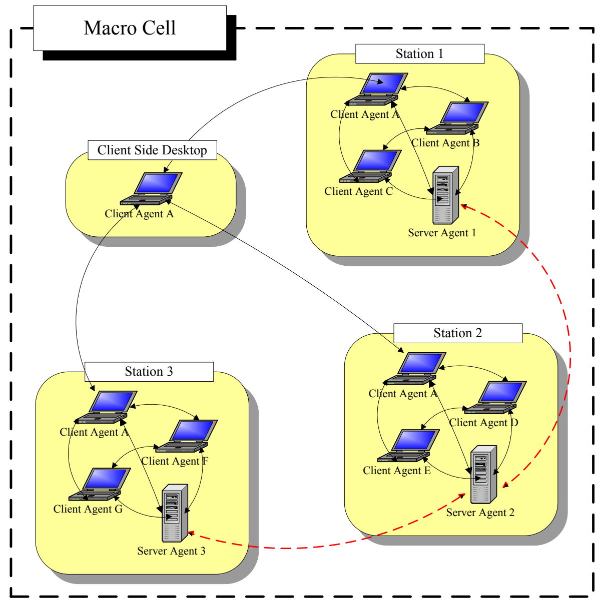


Figure 2. A Mobile Agent Society Environment

Server Agent:

The server agent contains four components: verification components, communication components, management components, and roles setting objects (agent characteristics) provider. As with non-mobile agent, the primary requirement is a method of delegating authority to the mobile agent.

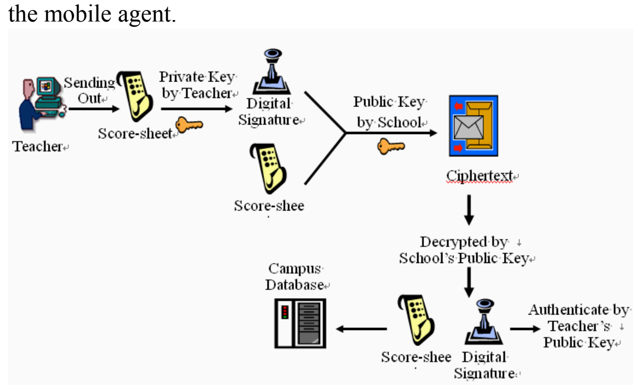


Figure 3: Verification processes

- Verification components: provide the security-minded with agent delegation and authentication, privacy and access control. (the processes of verification in case of updating the student's degree are shown as figure 3)

- Communication components: provide the universal communication tools, such as the chat (text), audio, video and windows message (annotation) application tools.
- Management components: provide the system management facilities, such as the administration, resource allocation and agent profiles modification functions.
- Roles setting objects provider: provide the application-driven/characteristic objects to the client agent to download those objects. (e.g. E-notebook, Authoring tool (for course design), questionnaire sub-system, lecture- on-demand and so on.)

3.2 Pico Cell

The client agent is the base unit (Pico cell) of the mobile agent society. The main elements of a client agent include the client profiles, object function loader and role setting components selector.

- Client profiles: User personal environment setting information, User Log Files, User personal information and User submitting results. (Those are same as the server side's profiles/database)
- Role setting components selector: this component provide several of role templates for user, user can choose the agent's role which represents the agent will possess some application tools and put to use in the society.
- Object function loader: after user had selected their role sets, the object function loader will download the related objects from server agent (Roles setting objects provider).

3.3 Operation

This section shows several examples of Mobile Agent operations. The examples illustrated three main phrases: (1) registration (2) communication (3) role setting.

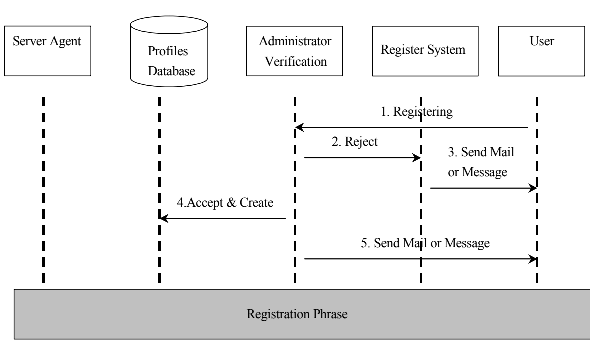


Figure 4. Registration operating phase

When a user wants applying into a new society, the client agent is invoked to coordinate the updates between the server and subscriber. Figure 4 shows the operations for a mobile registration to be a new society's member.

After the MMVS register system received the uses apply, it send a registration notification message to the administrator verification component (event 1, Figure 4). In this example (event 2), the administrator verification component sends a reject notification to the MMVS register system. Upon receiving response form administrator verification, the MMVS register system send a response back to the user (shown in the Figure 4 as event 3).

In event 4, the administrator verification records the registration acceptation message into the profile database. Event 5 is the acceptation response back to the user.

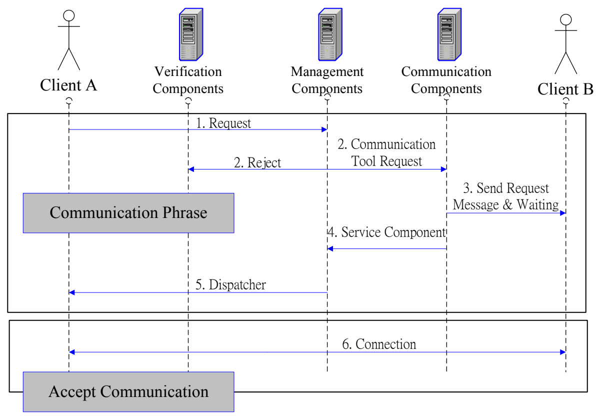


Figure 5. Communication phrase

Figure 5 illustrated an example of an invoking communication between two mobile agents. Event 1 is the request message to be sent to the management component of the server station. In turn, after the system's management component agrees this apply, it will send the communication tools needed and requests to the communication tool manager (event 2), this invoke message is forwarded to the destined agent (event 3). If the destined agent assents to communicate with the requested one, she will respond to the communication components manager and the desiring service component will sent to the system's management component (event 4). Thereafter, a connection pre-processing is established as depicted in this figure. The management components will dispatch the service component to original agent, and then accomplishing the connection between two agents.

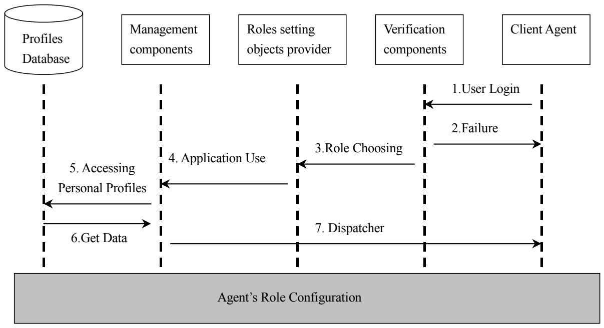


Figure 6. Agent's role setting phase

In the event that the client agent applies the roles setting objects, the server provides a set of commensurable application-driven components for the authorized agent. These operations were shown in Figure 6. The first two events handled the testing and verifying the agent status, and rejected the unauthorized login agents. The function of these events provided the delegation and authentication. Event 3 and 4 provided the role configurations. After the role configuration steps, event 5, 6 and 7 will record the user's profile and dispatch the desired agent role's objects to the user.

4. Discussion: Limitations and Solutions

There were many focus issues occurred after the system constructed. We had received many diverse opinions such as plenty responses, valuable suggestions and important request from teachers and staffs' feedback, although this made us much more challenge and difficulty when the system developing. But, for the system to implement certainly that the involution of administration's authority was necessarily and essentially. We summarized some important issues as followings:

1. For the principals of the equitableness, publics and reasonableness in teaching's evaluation. The teacher should provide completed information of its score record that in his course responsibly. This standard might reduce some disputation which caused of incomplete score record.
2. The System provided a Web page with fixed style or format (it combined with Ordinary score, Mid-term score, Final-term score and Semester score), and the data imported from the selected course database would be Course's Name, Class's Title, Teacher's Name, Student's List and so on. There was a convenient way to let teacher filled out the Mid-term score and Final-term score in his familiar software tools to evaluate the grade sequentially, and import those manipulative scores into the system through Web page. This approach would reduce the perplexity of direct output that was in various software tools, distinct number of record field, and dissimilarity of the field name.
3. There was an adjustable function for user that he could modulate the percentage of each item in Semester score calculation. It was possible to adjust one specified record/field with a reference explanation individually. Also it was able to get a static analysis on-line, analyze class grade data; of course, printing and saving functions were available too. It was complete record if the score data and other information were uploaded together which were the score data of all course, relative percentage value, exceptional processing, explanation of static analysis, etc.
4. It was encourage to all teachers to utilize the Grade Uploading System (GUS) in the promotion phase basically. There was a gift of IC Card Reader to come with the application of Natural Person Certificate, it also had several training course in Campus, and the Operation Manual in book and on-line learning were available at same time. There was a fully service that could satisfy to all teachers to increase their aspiration in GUS project.

5. Conclusions

In our experiment, it was possible to provide a fundamental for extensive application with identity authentication, non-repudiation and the information exchange was confidential and integrity. An agent-based mobility management system was introduced and the ways of constructing the application-driven mobile agent was addressed. We suggested a framework to model the mobile agent virtual society, which contained both mobile agent communication network and mobile agent evolution states. This approaches aspired to provide to the software developers who could get advantages in the agent computing and the management routine work. Also, the role-setting components are object-oriented approach. This approach not only gives the flexibly but also scalable in user's utility tools. In out experiments, the application-driven mobile agent actually improved the persistent look-and-feel for roaming student in distance learning environment. The Grade Uploading System in SJSMIT Campus has been a trial test since January 2004, and it also has started using on April 2004. There were hundred of course in the system at beginning, and the consequence got appreciative criticisms from teachers and staffs. We hope that, this study should be prolonged and applied to future communication network environment.

References

[1] Andrew Nash, William Duane, Celia Joseph, Derek Brink; "PKI: Implementing and Managing E-Security"; Osborne / McGraw-Hill, New York, 2001.

[2] Carl Ellison and Bruce Schneier, "Ten Risks of PKI: What're Not Being Told about Public Key Infrastructure", http://www.goci.com/PKIrisks.htm

[3] Russ Hously, Tim Polk, "Planning for PKI: Best Practices Guide for Developing Public Key Infrastructure", John Wiley & Sons, Inc., 2001.

[4] William Burr, "Federal Public Infrastructure Technical Specification: Part C – Concepts of Operations", TWG-95-108, National Institutes of Standard and Technology, Nov. 1995, http://csrd.nist.gov/pki/twg
[5] Alper, C. and Colin, H., Agent Sourcebook: A Complete Guide to Desktop, Internet, and Intranet Agents, Wiley Computer Publishing,, (1997).
[6] Dong-liang Lee, Lawrence Y. Deng, Ching Hsu Chan, Chin-Yung Yu, "The Applications of IA for the Emergency Systems", Proceedings of DMS2004-The 10th International Conference on Distributed Multimedia Systems, (2004), pp.461-466.
[7] L. Deng, N. Tang, D.-L. Lee, C. Wang, M. Lu, "Vision Based Adaptive Traffic Signal Control System Development, Proceedings of 19th International Conference on Advanced Information Networking and Applications (AINA2005), pp. 385-388.
[8] Diffie, W. and Hellman, M.E., "New Directions in Cryptography", IEEE Transactions on Information Theory, 22(1976), pp. 644-654.
[9] Rivest, R. Shamir, A. and Adleman, L., "A Method for Obtaining Digital Signatures and Public Key Cryptosystems", Communications of the ACM, 21(1978), pp. 120-126.
[10] B. Schneier, "Applied Cryptography", 2nd ed., Wiley, 1996.
[11] http://csrc.nist.gov/pki/panel/overview.html
[12] Chanda, D. and Martin, F., "Information Agents for Automated Browsing," Proceedings of the 1996 ACM CIKM Conference (CIKM'96), Rockville, MD, U. S. A., (1996), pp. 296-305.
[13] Finin, T., R, Fritzson, McKay, D., and McEntire, "KQML as an Agent Communication Language," Proceeding of the Third International Conference on Information and Knowledge Management (CIKM), (1994),.pp. 456-463
[14] Gosling and McGilton, "The Java Language Environment: A White Paper," Sun Microsystems, Mountain View, CA, U.S.A., (1995).
[15] Microsoft Corporation and Digital Equipment Corporation — http://www.microsoft.com, the Component Object Model Specification, (1995).
[16] Ousterhout, J. K., "Tcl: A Universal Scripting Language," http://www.smli.com/research/ tcl.
[17] White, J. E., "A Introduction to Safety and Security in Telscript", http://www.genmagic. Com

RFID Facilitating Strategy Using Taiwan's Distribution Center as an Example

Chang-Ching Lin*, Shien-Chii Shieh**, Wei-Hsiu Huang*

*Department of Industrial Engineering and Management, St. John's and St. Mary's Institute of Technology
No. 499, Sec 4, Tamking Rd., Tamshui, Taipei, Taiwan
tclin@mail.sjsmit.edu.tw
**Department of Industry Management, Tung Nan Institute of Technology
No. 152, Sec. 3, PeiShen Rd., ShenKeng, Taipei, 222, Taiwan

Abstract—*The applications of RFID are the most promising issue which have caught attentions from every aspects of the global supply chain members. Both Wal-Mart and the Department of Defense, USA have announced that they will be facilitating RFID technology as their mainly strategy of supply chain management. Top suppliers are asked to furnish the utilization of this emerging technology. The goal has been set up to catch the potential benefits of RFID which including enhanced products visibility .traceability and safety at the manufacturing and distribution stages even throughout the total lifecycle. More and more fields take aggressive steps on RFID adoption and some business cases have revealed the advantage of RFID technology.*

As an important manufacturing and distribution base on the world, Taiwan has noticed this trend. In order to promote the competitiveness and to link up with the world through the application of RFID, this research focus on the facilitating strategy that how a Taiwan's business implement a new technology. We take distribution centers as our observe target, the collaborate module of government and enterprise will also be discussed.

Index Terms*—Radio Frequency Identification (RFID), Distribution Center (DC)*

I. INTRODUCTION

RFID identification system is mainly and jointly composed of Tag, Antenna, Controller, Middleware, and Database [1,2]. Signal is delivered by IC within Tag that is not larger than 2 centimeter squares through Reader to the electromagnetic circuit power supply within IC Drive. Then Antenna transmits the information stored inside the memory to controller and via

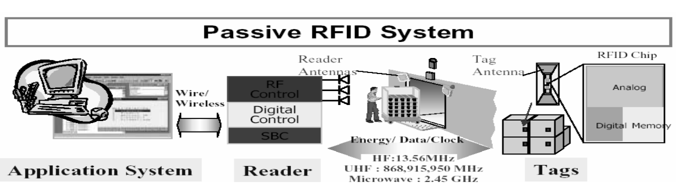


Figure 1. RFID composition Figure (Picture sources: Center For Aerospace and Systems Technology, ITRI)

Middleware encoding, ID Code is finally corresponding to adequate Database and is shown in Tag Identification System which characters that are human identifiable, such as shown in Figure 1— that is the construction figure of RFID. It also has the same contactless R/W properties as smart card and IC card, which can be used to replace the existing fixed or contact identification Tag [4]. And at the same time, it can read many data [2] and have better R/W ability and larger data storage space than Bar code.

In conception, besides hardware system, RFID should further include the corresponding software, and integrate the supply chain of enterprise both internally and externally. Figure 2 displays the complete framework of RFID system, which should include Tags, Readers, Integration, and Processing [5]. In Tag stage, to determine hardware specifications, we should depend on the environment estimation and analysis in advance, and then after the Tag is decided, the performance of reader & writer should be considered in designing the optimal position.

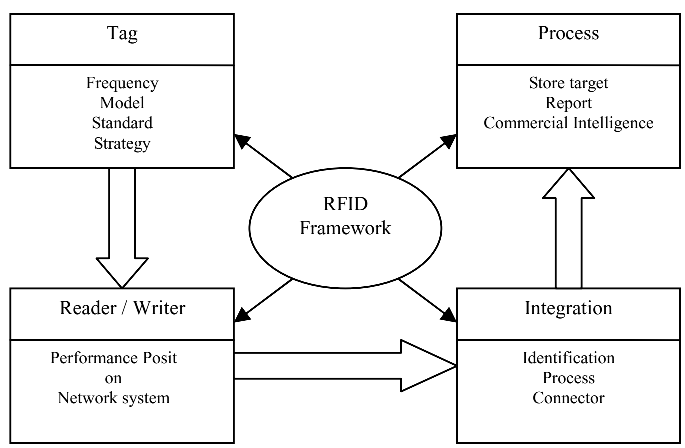


Figure 2. Framework of RFID system

And the network system, which connects reader & writer to database through wired or wireless method, analyzes the data through the accounting or tool software of stock logistics that is installed in the system beforehand, and stores it and produces it into each report for the analysis reference of decision maker.

However, RFID presently faces the largest bottleneck, that is, the specifications is failed to standardize. At present, two large organizations, namely EPCglobal of England and United States system, and Ubiquitous ID Center (UID) of Japan system have the most influence in the aspect of specifications promotion [4]. Recently, the UHF Generation 2 that is developed by more than 60 first technology companies of the world integrates RFID technology, and network through EPC network to provide the accurate information that conforms to cost-benefit in the supply chain. The so-called GEN2 standard is the basic element of EPCglobal network. In the future, they will look for the approval of International Organization for Standardization and it will compatible and collateral with the announced EPC standard, such as Class1, Class2, or even ISO 18000-6A and 18000-6B to provide a safer and faster open standard [2].

II. THE REFORMATION OF SUPPLY CHAIN

In the past, supply chain is connected linearly. Most use mails, facsimiles or telephone method to pass on information through vertical integration, and with the Internet development and the computerization of supply chain, it gradually develops into collaborative nonlinear supply chain. But in the future, market will be heading to demand-driven production method, such as shown in figure 3. Large reformation of market mechanism will also affect the production, marketing, and the stock strategy of enterprise; therefore, enterprises should conform to the supply chain form of the future, which is demand-driven. RFID is exactly the basic technology that accomplishes demand-driven supply chain form.

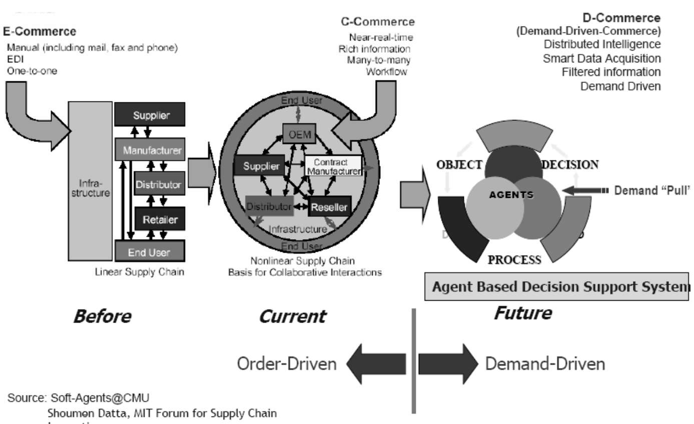


Fig 3. RFID development in promoting commercial mode

A. RFID will reconstruct the supply chain form

To satisfy the rapidly changing customers' demand, supply chain structure is also changing rapidly. Along with the growth of information in the whole world, enterprises should response to customers' demand with faster mechanism, and hence, enterprise flow path should be reconstructed. And in this process, RFID plays an important role, because the real-time of RFID and rich information can exactly conform to the rapid changes of demand, and at the same time, RFID makes supply chain be more observable and also let customers' demand be predictable. Therefore, we should depend on RFID technology to reach the supply chain that is mainly demand-driven. At present, RFID application is still in S&S（Slap & Ship）stage. Although it does not affect the operation method of present stage and peripheral cost does not increase due to the impact brought by RFID facilitating, the future demand-driven market is not really bearable to the present supply chain.

To be capable of producing many more yield and various production, the supply chain of future still need shorter time to reach and deliver the commodities with better quality. So besides enterprises' flow path reconstruction, Middleware is still needed to integrate electronic commercial affairs system, and the ERP and SCM system that is extensively applied these days should be integrated into the Middleware of RFID. The achievement and time demand of the abovementioned RFID characteristics can further predict the market in order to reduce the increasing cost and stock loss of enterprises due to the bullwhip effect of supply chain. These can only be achieved through the reintegration of RFID and enterprises electronic system.

B. Common problems of RFID facilitating into industries of Taiwan

At present, Taiwan industries often encounter some unconquerable problems in RFID facilitating, such as technology feasibility, hardware cost is excessively high, the estimation method of Return On Investment (ROI), the application of successful case, difficulties in facilitating cost estimation, deficiency of complete solution plan, the flow path facilitating method developing from S&S to internal part, and etc. Among which, ROI is one of the most important indexes for enterprises in executing investment evaluation, and today, since software is still in the developing stage, the cost is still inestimable. And due to the difference in application range, the specifications of RFID Tag are not completely the same and also, the cost increment caused by flow path reconstruction of Middleware and enterprises is beyond compute, so enterprises choose the wait-and-see attitude for the most part when RFID is introduced. ROI is often adopted by enterprises to be the reference index that judges whether to invest equipment or introduce new system. Author's individual viewpoint for ROI is that considering only the increasing benefit of RFID characteristic may neglect the ROI rough estimation value predicted by fixed cost. Here, we make the deduction and estimation as follows:

a. RFID can read many Tags at one time, so reading time（$T_{RFID} < T_{Bar\ code}$）

b. Real-time of RFID increases the visibility of supply chain, so inventory water line （$I_{RFID} < I_{Bar\ code}$）

The average profit growth rate of enterprise in the next n years:
$E_{RFID} = (T_{RFID} \times I_{RFID})^{-1} > E_{Bar\ code} = (T_{Bar\ code} \times I_{Bar\ code})^{-1}$

Return On Investment: (PER is Price-Earnings Ratio of enterprise)
$ROI_{RFID} = (1+E_{RFID}) \times n \div PER > ROI_{Bar\ code} = (1+E_{Bar\ code}) \times n \div PER$

From the abovementioned simulated calculation result, we can deduce that Return On Investment of RFID will far exceed the ROI of Bar Code with the application time of n years.

Table 1. The collection of phase target and achievements of individually counseled company

Company Name	Name of project	Plan range/purpose
Cargo Services Corp.	RFID application plan in Logistics	1. Read the Tag application plan on goods 2. The application plan of active Tag
Horng Woei Electronic Co., Ltd.	Feasibility evaluation and planning of UHF Tag safe control application inserted in tire	1. Design and develop RFID Tag inserted in tire 2. The test of each condition (temperature, pressure, material, and etc)
Silk road Taiwan Inc.	Feasibility evaluation of Savant middleware establishment in RFID handheld device	1. The study of PML & ONS 2. The study of handheld device 3. Savant test production
Goldsun Express & Logistics Co., Ltd.	Feasibility evaluation and planning of RFID application in the third party logistics service	1. RFID management condition in medicine industry supply chain 2. The key point of RFID in logistics operation management.
KHS Audio Co., Ltd.	RFID application in USA market and sale passageway of KHS Audio Co. Ltd, Jupiter Wind Instrument	1. Tag condition 2. Environment test 3. Tag evaluation 4. Reader collocation
Chiao Tai Logistics Corp.	Feasibility evaluation and planning of RFID integration application in 3PL supply chain	Feasibility evaluation and planning of RFID integration application in 3PL supply chain
Tai Sun Trade	Feasibility evaluation and planning of RFID application in professional cleansing industry	1. Accomplish RFID environmental plan 2. Accomplish the analysis document demanded by applied software 3. Learn about Tag & Reader function and installation 4. System integration result
Retail Support International	RFID Pilot Field Test in President Chain Store Corp. system	1. Assist logistics center to control logistics box and pallet stock quantity by means of RFID that is installed on logistics carrier container (logistics box, pallet) 2. Besides reducing the error risk occurrence in goods delivery, it can also well manage the goods.
Panda International transportation Co., Ltd.	RFID pilot logistics tracing test	1. Automatically report back the set off and reaching time. 2. Compile and analyze the statistics of transportation schedule. 3. Exception management. 4. Reduce man-made mistake. 5. Warehousing management.

III. The Related Facilitating Experience of Taiwan Industry Information

A. Strategy Framework of enterprise in facilitating ERP

In the early stages, industrial competition compelled enterprises to face a very big existence threat and forced all enterprise set competitiveness improvement as their target and for the middle-sized and large-sized enterprises with more than hundreds of employees, the facilitating of ERP system not only can integrate enterprise resources but also rapidly response to customer's demand changes. The strategy framework of enterprises when facilitating ERP can be divided into three large phases, namely diagnosis, design and facilitating.

And regarding facilitating flow path, it also can be subdivided into 5 steps, which are project preparation, business blueprint, realization, final preparation, go live & support respectively [3], such as shown in Figure 4. When enterprises facilitating ERP, its process is extremely similar to RFID facilitating condition of nowadays, and also faces the related problems concerning the supply chain integration of enterprise both internally and externally, which will be discussed in next section.

B. The problems faced by enterprises in facilitating ERP

In B2B E-Commerce application, the role that should be played by government and promoter is nothing more than the construction of trade environment, laws and regulations and establishment of trade standard. Therefore, government should construct a good network application environment to remove the existing obstacle, restrain in industry, and accelerate the revision of existing relevant laws and regulations, and establish new laws and decrees, intensify the domestic core enterprise counseling to secure commercial opportunities, and integrate resources in order to strengthen promotion achievements. And the problems that are often encountered by enterprises in facilitating ERP are as follows:

1. Fail to build good network application environment.
2. Deficiency of perfect laws and regulations of E-Commerce.
3. Middle-sized and small-sized enterprises still need to be subsidized and strengthened in education promotion.
4. Counseling should be aimed at industry, not specific enterprise.
5. Unsatisfactory integration in each promotion plan and unit resources.
6. Deficiency of professional information talents.

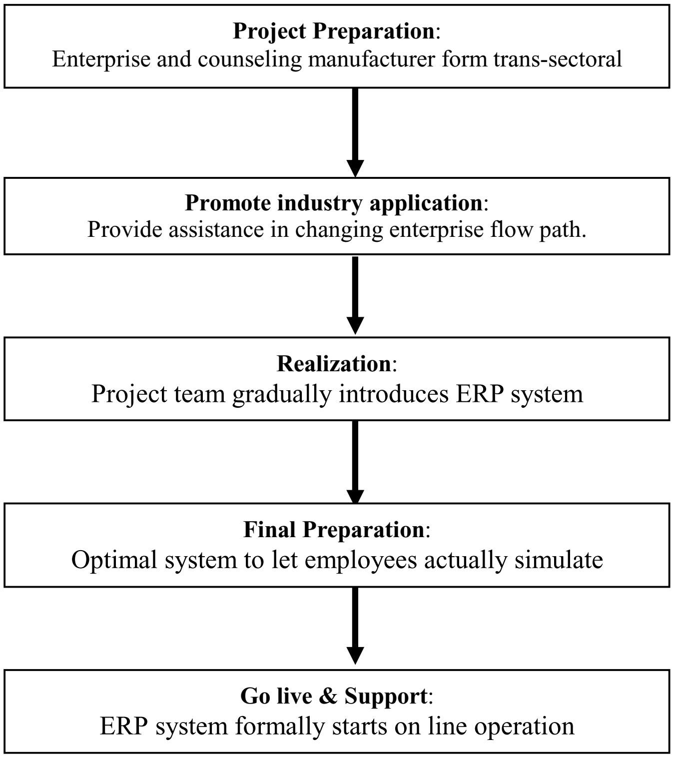


Figure 4. ERP Facilitating Procedures

In the abovementioned problems, the problems of item 3 and 6 are very similar to the problem encountered during RFID facilitating of present stage. Since the RFID of present stage is in pilot run stage, it still cannot provide effective exemplar construction to enterprise for reference, so most enterprises do not know the beneficial result of its facilitating, not to mention the related flow path disposition. And the authorities should also enhance the related study case with schools to increase the academic organization control and research of new technology and train up related professional talents to engage in the industry. Besides, the related technology seminar or workshop should extensively invite industry circle, authorities, and academic circle to jointly discuss it, and publish the result in responsible unit and provide window for enterprise consultation, and start improving environment (infrastructure) and related laws and regulations.

IV. Case analysis-A case study of Retail Support International

In 1979, President Chain Store Corp. established "Logistics Session", and in September 25th, 1990, the "Logistics Session" of the marketing department of President Chain Store Corp. was separated out, and four companies, namely Uni-President, President Chain Store Corp. and Mitsubishi Corp. of Japan, Ryosyoku Ltd. jointly put up capital to establish "Retail Support International", which is responsible for commodities logistics delivery business of more than 2300 shop fronts of President Chain Store Corp., PRESIDENT'S BAKERY, Cosmed and etc.

In 2004, Retail Support International was guided and counseled by MOEA of Ministry of Economic Affairs to lead the way in facilitating RFID to be hardware test (P1) and in 2005, they start to introduce the test of software and hardware system (P2). The future plan is to actually execute it to the internal of enterprise and construct RFID system exemplar after each department operates normally.

A. Case Discussion: Retail Support International

The target of RFID pilot field test project of President Chain Store Corp. system is to assist logistics center to control logistics box and pallet stock quantity by means of RFID that is installed on

logistics carrier container (logistics box, pallet). Besides reducing the error occurrence risk in goods delivery, it can also well manage the goods. Table 2 is the reading rate of pallet RFID at Retail Support International Center, table 3 is the reading rate

Table 2. Pallet reading rate

	Stock pallet number	Number of Tag read by Chungli DC	Chungli DC reading rate	Number of Tag read by Sanshia DC	Sanshia DC reading rate
Pallet	45	9	20.00%	3	6.67%
Plastic pallet	193	86	44.56%	91	47.15 %
Total	238	95	39.92%	94	39.50 %

Table 3. Logistics boxes reading rate at logistics center

Date	Actual logistics stock amount	Amount Read		Reading rate	
		Chungli	Sanshia	Chungli	Sanshia
Oct. 12	1351	1063	1162	78.68%	86.01%
Oct. 13	360	292	316	81.11%	87.78%
Oct. 13	1317	1050	1254	79.73%	~~95.22%~~
Oct. 15	1011	858	934	84.87%	92.38%
Oct. 15	779	851	698	~~109.24 %~~	89.60%
Oct. 18	1146	909	974	79.32%	84.99%
Oct. 18	828	768	779	92.75%	94.08%
Oct. 20	959	840	835	87.59%	87.07%
Oct. 20	720	629	646	87.36%	89.72%
Total	8471	7260	7598		
Average Value				83.93%	88.95%

Table 4. RFID reading rate of Chungli Ji-Li Store of President Chain Store Corp.（7/11）

Date	Actual logistics stock amount	Amount read at shop front	Reading rate
Oct 12	3	3	100%
Oct 13	6	6	100%
Total	9	9	100%

of logistics box RFID at Retail Support International Center, the RFID reading rate of table 2 and 3 are mainly based upon the test result of Taipei and Sanshia logistics with 8,471 logistics boxes tested, 193 plastic pallets, 45 pallets. From Table 4 RFID reading rate of Ji-Li Store, it is known that the RFID reading rate can reach 100% at stores, and the related problem to RFID reading rate would be discussed in the next section.

B. Generalization of Test Problem and Suggestion

1. Bad reading rate

Since test stage adopts manpower trolley to haul pallet, the condition of each reading is not exactly the same that causes unobvious reading effect. If it is changed into unmanned trolley hauling, each reading condition can be fixed [5,6] and further promote reading rate.

2. Tag repeated reading

Increasing the reading distance between Reader and Tag is no doubt contributive to the promotion of reading rate, but it also cause material and Tag that has not passed through Reader is read into system and cause system disorder easily. If the Tag is written into mark by Reader & Writer [4], or identifying the Tag read in Middleware and do not record the Tag with the same ID in the first record will avoid the repeated reading.

3. Reading Metal and Liquid Type Content

At present, RFID Tag is also subjected to the interference of metal and liquid, which cause identification cannot be correctly executed. Increasing the distance between Tag and logistics can no doubt reduce interference, but it also reduces the volume of logistics box and results in the increment of delivery cost. If insulation object (preventing signal interference) is loaded in the wall of logistics box, it can avoid the problem of metal and liquid interference [5,6].

4. Reading Condition

For goods piling on pallet, we adopt (width) 3 boxes * (depth) 3 boxes * (height) 4 boxes method, because in the past, they adopt Barcode reading Tag to execute warehousing management, piling height is too high and causes the Tag of goods within central area cannot be read easily. If (width) 3 boxes * (depth) 3 boxes * (height) 2 boxes piling method is adopted, the tag at central area can be effectively read, and besides, Reader can also be lodged in the upper part of Inspection Gate and floor to increase reading rate.

5. Suggested Items

Making comprehensive survey of the abovementioned reading problem, if carrier or logistics box reserved for RFID use is used, it can overcome the problem of bad reading rate easily. Besides, it should be matched with exclusive RFID reading area and connect to Database through Middleware. This complete set of measures can promote the whole effect of RFID, not only focusing on the reading accuracy with S&S method. Reading accuracy will be improved after technology is promoted, and the related whole set of integration introduced should all be tested in advance to obtain more objective information during facilitating to be the policy decision reference of facilitating in the future.

V. Comprehensive survey of RFID Facilitating Strategy

In November 2003, Wal-Mart formally announced to start facilitating RFID technology to manage and control the incoming and outgoing daily commodities in its logistics center. At the same time, it also requests its supplier to comprehensively use RFID since 2006 [3] which gradually makes related enterprises in the whole world face squarely to this application. This chain effect also facilitates each international or domestic large enterprise to forestall the facilitating of this technology in order to maintain or get ahead of its competitors. The success of the facilitating effect should be decided by the adequate facilitating strategy. This section will also compare the global strategy with the strategy developed in Taiwan.

A. The strategy of Taiwan Ministry of Economy Affairs.

To strengthen the international competitiveness of Taiwan enterprises, MOEA of Ministry of Economic Affairs especially cooperates with ITRI in “Industrial RFID Application Counseling Plan [2]”.The system center of ITRI is responsible for the coordination, operation and preparation affairs.

Besides technology facilitating, ITRI is also responsible to make a collective report of the progress to MOEA of Ministry of Economic Affairs, and holds nonscheduled achievements publication to be the reference for industries. And Institute for Information Industry is responsible for Middleware operation, and integrate the electronic platform of enterprise such as ERP with RFID, and stores the information read by RFID in ONS of network by using EPC encoding method through PML, which enables the server of enterprise to read ID information and further achieve globalization target. International R&D unit further publishes new technology and standard, and ITRI has the new information in hand at any time and provides the new technology to domestic semiconductor industries and information industries, while the enterprises use the new technology to develop the software and hardware with more competitive advantages.

When Ministry of Economic Affairs develops facilitating strategy, it can be divided into four stages, where the main purpose of P1 Field test is to test the hardware reliability, in another word, that is to test the read rate of Tag. Besides viewing the Tag, it also tests the impact of reading angle or position of Reader on reading rate. And the target of P2 Evaluation Test is to test the middleware.

The main purpose of this stage is to connect the database with Reader, to test the problems resulted from actual operation and improves them to enable enterprises to actually obtain the complete information of RFID Tag from database. In addition, P3 is Pilot Run stage. Besides integrating RFID Tag system, the authorized unit should further integrate E-Commerce system and supply chain management system to reach the anticipated effect of whole integration. The last is P4 Exemplified system construction stage to construct standard prototype, and the result is provided to be the reference for enterprises during facilitating RFID and provides the consultation service of related technology.

B. The RFID facilitating strategy of IBM.

Besides government guidance and counseling, International industry also actively participates in the study and plan of strategy facilitating in facilitating RFID to enterprises, where many industries provide a series of solution program. Besides developing a series of hardware technology, the related necessary middleware also tends to the integration of enterprise platform and supply chain, and even relevant enterprise that introduces

ERP also submits enterprise solution program in succession.

At first, they should carry out the lowest demand of case evaluation analysis, and replace Bar Code and promote the accuracy of existing application, and execute the optimization. In this stage, they should promote enterprise's productivity, and reduce the demand for manpower and drive JIT production or logistics. At the same time, real time strategy support should be started to provide internal stocks and personnel tracing and achieve the prompt procedures of automation. In the last stage, we should change the operation mode of enterprise, increase new profit and collaborate with external customers or supplier, and trace goods historical record. Eventually, commodities can be accurately traced both internally and externally through the value chain of Figure 5.

From the value chain framework of IBM, we learn that RFID Tag is attached immediately after commodities (products) are produced, and they are monitored with RFID during production line, and RFID Reader is also set in the stock area to actually read each item and quantity of stock. When commodities is delivered to logistics, RFID Reader of logistics center reads the Tag, and the commodities are delivered by forklift that is installed with Reader to storage area. At this time, the information will be sent back to server, and enterprise can compare the accuracy of stock quantity and can also gather statistics of the stock quantity at present for the convenience of adjusting procurement strategy. When stocks leaving factory, RFID reader of packing area will read the quantity of stocks leaving the factory. In addition, division area is also installed with Reader, and eventually, the commodities is delivered to stocks leaving area, and before stocks leaving warehouse, there is one more RFID Reader to once again report back the information to server. After goods leave logistics center, they will be delivered directly to customer's warehousing area. There is RFID Reader in the warehousing area, which can rapidly check over the items and quantity through the database connected by Middleware, and then goods covering unit on-site allows the commodities to pass through sensor gate of RFID Reader in arranged order to the display shelves of market. Furthermore, the display shelves are also installed with RFID Reader to monitor the inventory and sales condition, and lastly, customers can quickly pay up at the counter, which is installed with RFID Reader.

VI. Summary and Conclusions

RFID has made further progress under the promotion of Taiwan government with great exertion, but comprehensively surveying the progress of advanced countries such as Europe and United States, the pace is obviously slow. And besides the software and hardware promotion, the aspect that really influences the whole effect is strategy aspect. The facilitating strategy of RFID has already been decided since the beginning, where the strategic error not only is incapable of reaching the set target, but also wastes money and manpower, and enterprises' hanging back is also predictable. Hence, the feasibility of actual strategy will be discussed, and reference to successful examples for modification will be need at any time. In the following, we sort out the facilitating strategy that should be constructed to be the reference:

a. Develop the complete end-to-end solution program
   1. Simplify the facilitating process
   2. Reduce facilitating cost
   3. Reduce the facilitating risk and uncertainty
b. Establish counseling team
   1. Enterprise flow path analysis
   2. Draw up application range
   3. Provide assistance in evaluating ROI

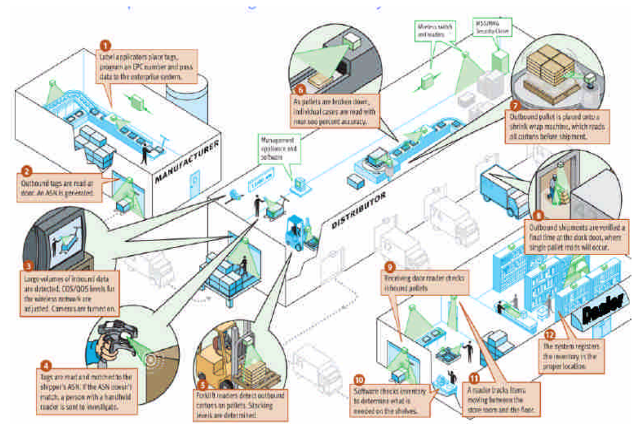


Figure 5. Value chain framework plan of IBM

   4. Assist enterprise flow path reconstruction
c. Establish Point Of Contact (POC) Center
   1. Feasibility analysis
   2. Define S.O.P. (Standard Operation Procedure)
   3. Provide assistance in choosing suitable HW (Hardware) / SW (Software) products.

Of course, RFID is not the substitute of Bar Code, but observing from each aspect, besides the ID identification provided by original Bar Code, RFID is even more superior to Bar Code in many aspect. Such as time shortening, reduction in manpower demand, value chain visibility and etc, are all a part of RFID big and powerful superiority. In the future, when RFID integrates supply chain into demand-driven formation, it even shows forth RFID value in supply chain.

References

[1] Vince Stanford., "Pervasive Computing Goes the Last Hundred Feet with RFID Systems," *IEEE Pervasive Computing*, 2004, pp. 9–14.

[2] "The Mission of The HAG is to Define The Interfaces Between Hardware Components (Primarily RFID Tags and Readers) in the EPCglobal Network," *EPCglobal*, 2005.

[3] Roy Want., "Enabling Ubiquitous Sensing with RFID," *Intel Research*, Vol. 2, 2003, pp. 2-5.

[4] Siden, J., Jonsson, P., Olsson, T., Wang, G., "Performance Degradation of RFID System Due to the Distortion in RFID Tag Antenna," *Microwave and Telecommunication Technology*, Vol. 11, 2001, pp. 371-373.

[5] Foster, P. R., & Burberry, R. A., "Antenna Problems in RFID Systems," *RFID Technology*, Vol. 123, 1999, pp. 1-5.

[6] Xingxin, G., Zhe, X., Hao, W., Jun, S., Jian, H., & Song, S., "An Approach to Security and Privacy of RFID System for Supply Chain," *IEEE International Conference*, 2004, pp. 1-5.

3C Intelligent Home Appliance Control System

Hsu Chun-Liang[1] Lawrence Y. Deng[2] Kou Cheng-Chien[3] Chen Po-Hung[4]

E.E Department of Saint John's Technology University[1, 3, 4]

C.S.I.E. Department of Saint John's Technology University[2]

+886-2-28013131 #6500, liang@mail.sjsmit.edu.tw

Saint John's Technology University

Abstract

Since Automation spread throughout the manufacturing in last decade, there seemed no such equal attention with home appliance. In this paper, the automation developed into the place where we spend most of our time becomes a brand new thinking. There are so many home appliances in our living space, and how to make them intelligent so as to let our life more saved, convenient, and comfortable will be a challenge to our knowledge. In this study, some technique and skills were developed to make our home appliances intelligent such as radio remote control, sensors invented by researchers, auto-dialing system, and image analysis with CCD image-fetched. Through linking the sub-systems' function, the main function of the home appliance worked smartly—in this case, we use a refrigerator as the center of all home appliances in a family, the refrigerator could sense the lack of the food, drinks, even irregular shape fruits through all designed sensors and CCD images analysis system, and it also could auto-dial the vendors to deliver the thing it sensed insufficient. In addition to detect capability, the refrigerator could assign works to other appliance through inter-net or wireless radio remote control system, for example, supposing that we forgot to turn off the gas oven after we had reached far away our home, we could dial cell-phone or phone into the refrigerator to generate command code to turn off the fire through remote control module set up both on refrigerator and oven. This design can insure our life saved and convenience.

Key words: **3C (computer, communication, content), Home Appliance, Intelligent, CCD (charged-coupled device)**

1. Introduction

Automation has spread over many fields such as manufacturing, offices, businesses, and buildings. But, it seemed that the place—our living space –home got no equal attention, especially home is the most time where we spend. Therefore, we try to take advantage of our knowledge of automation combined with web-net to make the home appliance become intelligent center that can play not only its original role but also control other appliances by wireless communication in home space instead of merely an appliance. Since refrigerator is most frequently used by housewives in a home, we want to invent an intelligent one which can automatically detect the quantities of materials in it such as milk bottles, drinks, soda bottles, eggs, bread, fruits, or the amount of irregular shape such as vegetables, meats etc.. If the quantities or amount are less than we expectation, the refrigerator will call the supplier by web or auto-dialing system to ask them deliver the thing we need. This mechanism can create more convenient and comfortable life for busy modern people. In addition to refrigerator original function, we created it an intelligent function to control the other appliances in same living space of a home. This function makes a refrigerator no longer be an appliance, furthermore be a control center which will give order to other appliance after getting messages through web-net or phone auto-dialing system operated by man outside the home far away. For example, we all have such awful experience as forgetting turning off gas oven and remembering it when we are far away home on high way, and we are panic and helpless. If we had one refrigerator owning function like controlling other appliance—in this case –the gas oven by dialing a command code through wired phone or cell phone, we can easily turn off the gas over so as to prevent miserable fire disaster.

2. Study purpose

(1) Combining life with automation conception to humanize appliance with knowledge and make people feel the comfort and convenience bringing by technology.

(2) Making home appliances simple operation, systematization, computerization, and information oriental so as to get appliance work with more efficiency, intelligence, convenience, and safe.

(3) Finding an optimal solution for combining computer, wireless communication system, and

microprocessor into a full-function system to upgrade the value of single system.

3. Design procedures

Since we want to design an automation & intelligent appliance with 3C function here we take example by a refrigerator, we expect the refrigerator to have function as follow:

(1). Auto-detecting the quantities of normal shape objects in it such as bottles, cans, fruits, and eggs or amount of inform shape objects such vegetables, meats, and fishes. The auto-detecting system would consist of all kinds of sensors which could collect the quantities and amount information of objects in refrigerator. For example, we use photo-coupled transistors to detect the bottles, the pressure plate to sense the amount of meat or fishes, principle of closed-tub pressure to sense the amount of liquid in big plastic bottle like soda water and milk, and CCD images fetch to detect the inform objects like fruits.

(2). If the objects are detected less than we expected, there would be an auto-dialing system through modem to call the supplier to deliver the objects we need so as well the shortage message would be displayed on the screen on computer. Therefore, we need to design an auto-dialing system based on PC computer in order to ask the supplier to deliver the things we need after detecting the things are less than we expect.

(3). Because the refrigerator is the intelligent facility, it will give commander to the other appliance through wireless remote control system and in this practical study was making use of single chip microprocessor 8051 combined with Appliance Automation concept. The user can easily control the appliances' "on" or "off" in a house by pushing a keyboard button that was in advance corresponded to an appliance either in a house or far away outside the house. The device was worked by a system with two control modules called A module and B module; the former was responsible for decoding the dialed number either from the keyboard on A control module board beside house phones or from the phone's keyboard dialed number far away outdoors, and then A module would send the "controlling code", which would order some appliance "on" or "off" after B module receive the controlling code, to B module by high frequency microwave. That means there were wireless remote controlled subsystem devices on both A and B modules. The device has been proved high stability in itself and is able to become a mass-production product in manufactured business. So we need to design a wireless remote control system which consists of two module, module A is connected to the phone to receive the command code from outdoors under master's request. Module B is connected to others home appliance which will receive the command code from module A to turn on or off the appliance we request. In this case, we take the example by a gas oven as the controlled appliance which would accept the commander from the refrigerator. Supposing that we forgot to turn off the oven and we suddenly remembered the terrible situation when we was far away home for our vacation after we left the house. We don't need to worry about this, because we just dial a "command code" by cell phone to module A on the refrigerator then through module B to turn off the gas oven.

(4). We can easily detect the normal shape objects like bottles etc., but how to detect the informal shape things like all kind of fruit becomes a difficult issue. Therefore, we use CCD to fetch the images of informal shape things and transfer into specified data to be compared with the data base we set up beforehand for those informal shape things, and then classified them by color featured value. The analysis procedure is shown as figure 1 and 2.

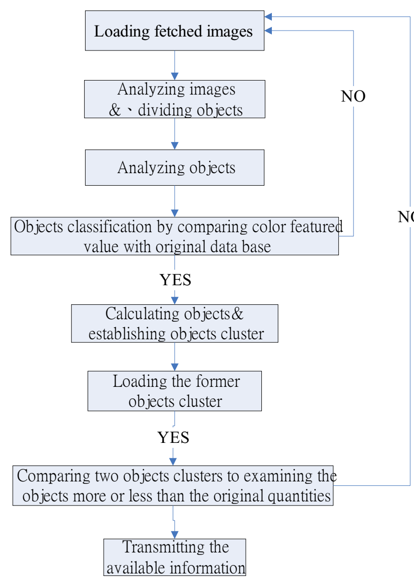


Figure1 Images analysis system flow chat

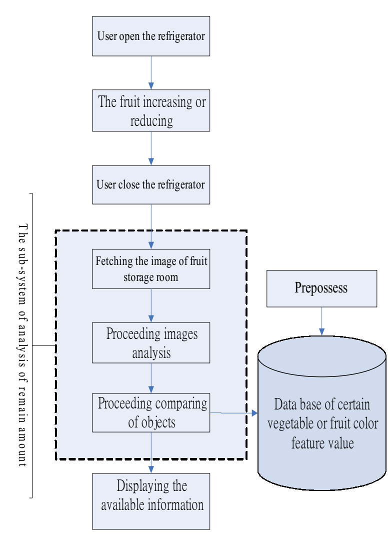


Figure2 CCD images analysis flow chat

(5). Finally, we must design the PC bases Graph Interface System as the platform to give command or monitor the information of refrigerator, here we use Visual Basic language to design the interface platform and the whole 3C intelligent home appliance control system is shown as figure 3.

4. Conclusion

Automation has been a new trend in many fields in the last decades, and many program and curriculums are designed to meet the demand in the vocational markets. In this study, we try to initiate a new think of home appliance—not only did we consider appliance to be useful family devices, but we increase the benefits of appliance after they are gifted intelligence, and this is available if we add more performance of automation knowledge to them. The study procedure was difficult and hard, but the final outcome was more delighted after the product was proved totally successful, and we really realized the true meaning that knowledge creates the extra economical profits. This most outstanding outcome of this study is that the CCD images analysis technique in refrigerator has gained a patent issued by Taiwan official patent assessment bureau.

References

Tsai C.Y. [2003]. *Theory & Implementation of Single Processors 8051.* Chuang-hwa Press IncTaipei.

Deng G.C. [2003] *Projects Research of 8051 Single Chip*. Hong-yu Press Inc., Taipei.

Gu Y.Z. [2003] *Technique Book of Visual Basic 6.0*. Gi-Phong Press Inc. Taipei.

Huang M.D.[2003] *The Examples and Analysis of Visual Basic 6.0*. Chuang-hwa Press IncTaipei

Wang K.R. [2003] *The Database Programming Design*. Chi-biouw Press IncTaipei.

Fan Y. Z. [2002] *The Serial Communication Interface Design of RS-232 with Visual Basic*. Wen-quei Press Inc. Taipei

Cheng H.Z. [2005] *The Study of Images Search Technique Based on Color Tree*.,Taipei

Liu Y.T. [2005] *The Images Searching System Based on Feature Value of Color of Single Objec,* Taipei.

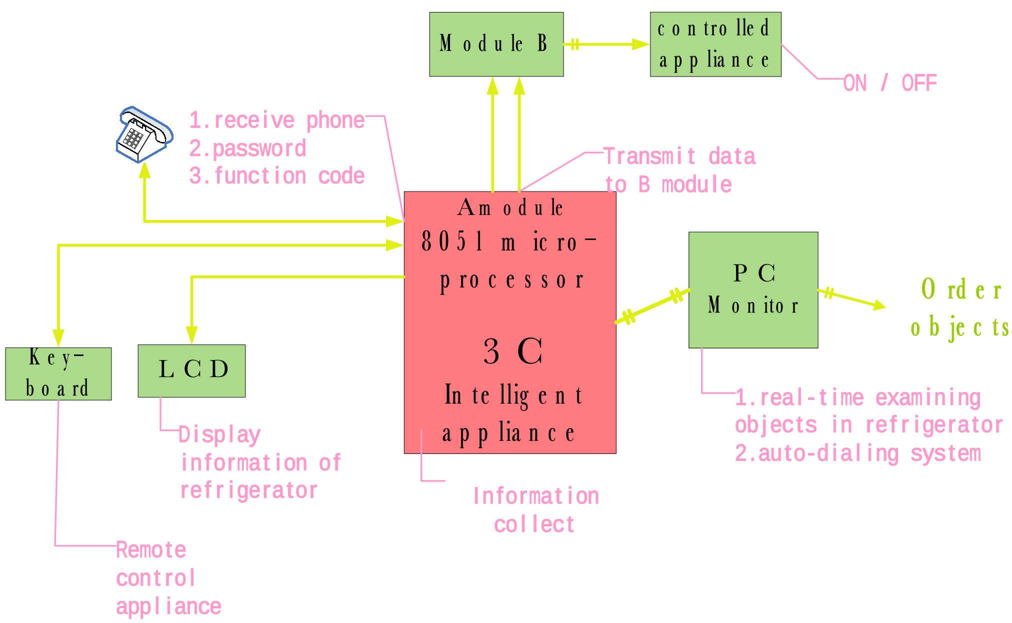


Figure 3 conceptual structure of 3C intelligent

Depicting the Algorithm of Discrete Topology for Distance Learning System

Chih-Yung Chen[1] Jih-Fu Tu[2] Chun-liang Hsu[3]

[1]Department of Information Management
[2]Department of Electronic Engineering
[3]Department of Electrical Engineering
St. John's & St. Mary's Institute of Technology
{yung, tu }@mail.sjsmit.edu.tw

Abstract

For improving the performance of distance learning system (DES), the communication protocol between clients and server is very important and has much space to improve. In this paper, we issued the protocol and algorithms of communication, and data-receive and data-send to implement high-speed data transmission between processors of the DES to archive and apply the distance learning systems, respectively.

Keywords: Distance learning system communication network, discrete topology, and communication handshaking.

1. Introduction

The quality of the discrete computer system (DCS), also called tightly coupled system, has a decisive impact on the speed, size and cost of the whole architecture. In discrete computer system any processors must be able to access any memory location, even if it physically belongs to another processor. Thus, the communication interconnection schemes are usually employed. The communication machine for discrete computer system allows the messages to be transmitted between execution units of different processor [4,5,8].

The communication interconnection network can be further classified into two types: signal bus and multiple bus systems depending on the number of buses providing interconnection among the processors, called node, of the discrete computer system architectures.

In this paper, we propose a communication protocol for the multiple buses of DCS. The communication latency time is the one of benchmarks of interconnecting processors of DCS [5,7]. The communication delay time is defined to *the total time of the request is issued to the desired data are received*. In fact, the communication latency imposes a limiting factor on the scalability of the communication mechanism size. Thus, the number of processors of DCS cannot be increased indefinitely without exceeding the tolerance level of the access latency. For a DCS, the desired data of a processor may be accessed through two sources, the local memory (called P-to-M or P2M) or other processor (called P-to-P or P2P) [1].

A communication network can be represented by a communication graph. The topology of the communication graph is an important property, which significantly influences latency on multiprocessors. Thus, there are two-communication models, message-passing model and share-memory model, are discussed and analyzed in this paper. For the message-passing model, the information is exchange through an inter-process communication facility provided by the communication routine [2,3]. In the share-memory model, processes use map memory scheme to gain access to regions of memory owned by other processors.

The recent of this paper is organized as follows. Section 2 describes the protocol of communication for DCS. The communication latency and performance of DCS is analyzed in Section 3. Finally, we remark the conclusions in Section 4.

2. The Protocol of Communication

A DCS is a collection of tightly coupled processors interconnected by a communication network. From this point of view of a specific processor in a DCS, the memory is divided several modules and shared among processors, the rest of the processors and their respective resources, such as data cache, are remote. The designer of a communication network must address four basic issues:

1) Naming and name sources: such as controller, processors, and storage devices, how to process and locate each other to communication?
2) Routing strategies: how are messages sent through the network?
3) Collecting strategies: how do two processes send a sequence of messages?
4) Contention: the network is a shared resource, so how do we resolve conflicting demands for its use?

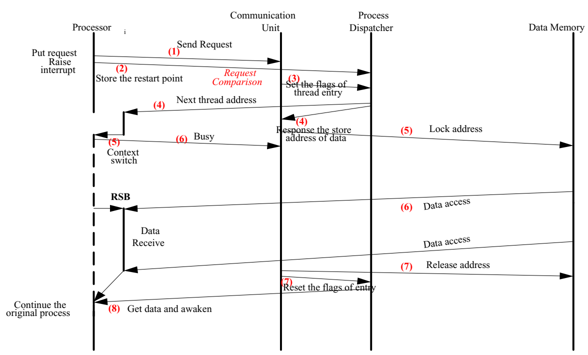
Communication control can take place only if the correct processor access rights are granted. Each processor has a corresponded message queue, which holds a sent message, in the bus decoder of this paper. When the message queue is filled, the sequentially messages are stalled by this message queue. In the communication network architecture, the processors are classified into the requesting processor and the requested processor. The requesting processor, also called receiver, can do one of the follows things:

1) receiver can send the request to the bus arbiter

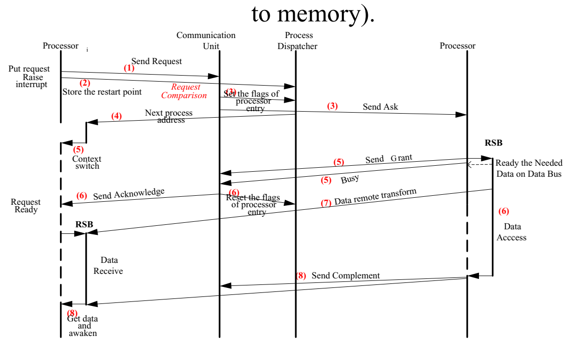
while data dependency occurs,

2) receiver can block until the message is received successfully,
3) receiver can ask the bus decoder to hold it temporarily, and
4) receiver can give an error message while the desired data losses.

For one processor of the DCS will get data from other processors. Firstly, The requesting processor must send a request message, which contains the address of the desired data, to the bus arbiter, then to the bus decoder, furthermore to the requested processor. The request message is transmitted to the requested processor via the communication network. If the requested processor is busing, the request message is blocked in the request message queue until the requested processor is free.



(a) The communication protocol of the P2M (processor to memory).



. (b) The communication protocol of the P2P (processor to processor).

Fig. 1. Communication protocol of DCS.

The requested processor receives the request message; the requested processor must send a reply signal to the requesting processor. The above process among processors is called handshaking of communication. Implement the handshaking process, then the desired data is transmitted from/to processor through the data queue of the bus decoder. The desired data of the requesting processor may be transmitted from the shared memory or the requested processor, which are called p2M (processor-to-memory) or p2p (processor-to-processor), respectively. The communication protocol of the p2p and p2M are illustrated in Figure 1(a) and Figure 1(b), respectively. The algorithm of communication process of the DCS for this paper is illustrated in Figure 2, too.

The communication machine of this paper is the combination of bus arbiter and bus decoder. The bus decoder is constructed of request buffer for buffering the request of requesting processor, data buffer for storing the desired data from the requested processor, request message register for registering the request when the requested processor is block or busy, and the controller for initializing the requested processor to access the desired data from its local memory. This communication machine allows transmitting the message and data between the executable units of processors.

The structure of the request and data buffer are FIFO (first in first out) architecture to hold the request message and the desired data of the requesting processor. In practice, the buffers can only be a finite length and a point could be reached when the contents are help up because all the available buffer space has been exhausted.

The algorithm of the communication protocol of DCS is shown in Figure 2. For a receive routine, the desired data has been received of the requesting processor if it has been sent to the data buffer of the communication unit. If the receive routine of the requesting processor reaches before the send routine of the desired processor, the data buffer of the communication will be empty and receive routine must wait until the desired data is sent to the data buffer. For the send routine of the desired processor, once the local actions have been completed and the desired data is safely accessed on its way, the process of the requesting processor can continue with the subsequent work and instruction. In this way, we use such receive and send routines interaction between requesting and desired processor to decrease the overhead of the communication time

In this paper, we will only discuss with the communication protocol of the processor-to-processor (p2p). Let us examine the basic message parameters associated with sender and receiver system calls. The message-passing primitives are specified by:

- Create and delete the request buffer and data buffer of the communication unit via the *request* and *data_receiver* routines.
- Temporally, storing the transfer status information of requesting processor to the request buffer of the communication unit.
- Send the request signal and receiving the desired data among processors,
- Attach and detach remote processors, such as requesting and requested processors

```
/* The algorithm of interconnection processor communication network*/
Repeat:
:
requesting processor has an request item in next-tp of the request buffer of communication unit;
:
/* sending the request message to the desired processor */
send (desired processor, next-tp);
until false;
the requested processor process is defined as the needed for the requesting processor;

/*receiving the desired data from the data buffer for the requesting processor */
/* receive routine*/
receive (requesting processor, next-d);
:
requested processor send the data item to the next-d of data buffer;
:
```

Fig. 2. The algorithm of inter-processor.

The send and receive primitives of the communication unit are implemented by the sending and receiving process, respectively. Therefore, the buffer field in send specifies the memory location of the data to be retrieved from the requested processor. On the other hand, the buffer field in receive specifies where the arriving data will be stored before to be read by the requesting processor.

. Figure 2 illustrates the communication topology for the IPCN. The data is transferred between inter-processor to be completed by the *send ( )* program for sending the needed from the desired processor to data buffer of communication unit, the algorithm of *send ( )* is sown in Figure 3. The *recv ( )* program for receiving the needed from the data buffer of communication unit to requesting processor, the algorithm of *recv ( )* is sown in Figure 4.

```
/* data transmission */
    send ( )
        {
            data_transmitter:
                wait:
                if (data is ready for the requested processor and received the signal of request)
                    { data_id is set to active;
                        access the desired data from local memory of GPR of requested
processor;
                        send the desired data to data buffer of the communication unit;
                        data_ip = data_ip+1;
                            if (data_ip != 0)
                                goto data_transmitter;
                            else end;
                    } else wait;
                end:
                data_id is reset to valid;
}
```

Fig. 3. The algorithm of data send.

```
/* data receive */
        recv ( )
            {
                data_receive:
                    wait:
                    if (data queues in the data queue of communication)
                        {
                            the flag of data_valid is set to 1;
                            the signal of the received is ready for the requesting processor;
                            receive the desired data from data buffer of the communication unit;
                            forward the received data to the general-purpose register (GPR);
                            data_ip = data_ip-1;
                                if (data_ip!= 0)
                                    goto data_receiver;
                                else end;
                        }
                    else wait;
                    end:
                    the flag of data_valid is reset to 0;
    }
```

Fig. 4. The algorithm of data receive.

The structure of this proposed DCS is shown in Figure 5, which is consisted of n numbers of processors, m numbers of memory-modules, b numbers of buses, and the communication unit, included the bus arbitration and the bus decoder.

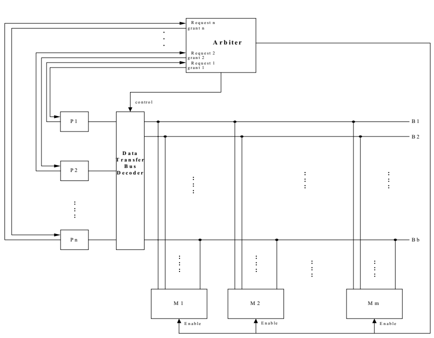


Fig. 5. The schema of DCS for this paper.

The desired data of the requesting processor may access from one of two paths, other processor via remote produce call (RPC) or shared memory (memory module) via direct-access memory (DMA). In this paper, we only discussed with the data transmission between processor to processor. Thus, we also represented the state transfer machine of the p2p in Figure 6.

If the desired data is accessed from other processor (i.e. requested processor), the communication handshaking starts the place of *request data from other* for the requesting processor*,* then initialize to the transition of *request queue*, furthermore feed to the place of *request queue.* If the required processor is busy, the request is temperately queued in the place of *request register*; else the required processor is initiated the transition of *request interrupt* to access the desired data via the transition of *system call*.

The desired data are accessed to the requesting processor from the required processor through the transition of received data and the place of data queue. If the required processor does not generate the desired data, the respondent signal of the request is still blocked in the place of ready to receive data, until the requested processor completes the desired data.

3. Performance Analysis

The communication latency is the one of performances of inter-processors architecture. We will analyze the communication latency of this proposed architecture for different number of clients with different size of the request buffer and data buffer. The number of request- and data-buffer are various from 2 entries to 16 entries.

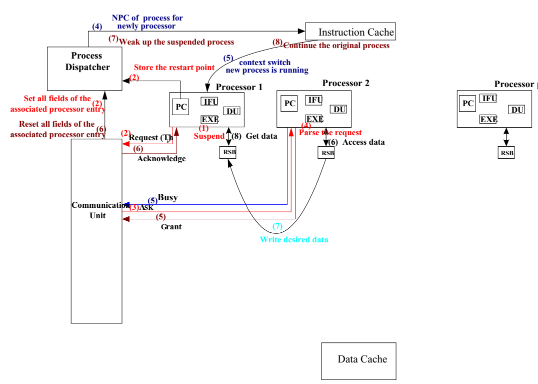


Fig. 6. The data transmission between processor and processor.

3.1 The Communication Latency of Different Queue-Size

In this section, we compare the communication latency of this proposed DCS in the following cases: 1) the proposed MP has 6 PEs, but with different numbers of request- and data-buffer, 2) the proposed MP has the same size of request- and data-queue, but with different numbers of PE.

Figure 7 shows the communication latency of different numbers of request-buffer in case 1. Referring to Figure 7, we find the lowest *latency to cost* ratio, is defined to the buffer size is divided to the reduced time ($T_{latency}$), of each PE is 8-entry, in average. This reason is caused that the request is issued from requesting processor to communication unit, then the communication unit issues an interrupt single to the requested processor, Thus, any requests must expended waiting time in the communication unit. When the request queue size large than 8-entry, such as 12 or 16 entries, the latency time not conspicuously improve for the cost.

Figure 8 shows the communication latency of different numbers of data-buffer in case 1. Referring to Figure 8, we find the lowest *latency to cost* ratio occurrence in 4-entry. This reason is caused that the desired data of the requesting processing is accessed from required processor to the data-queue of communication unit, then transmitting to requesting processor, immediately. Thus, the desired data not expend any waiting time in the data queue of communication unit.

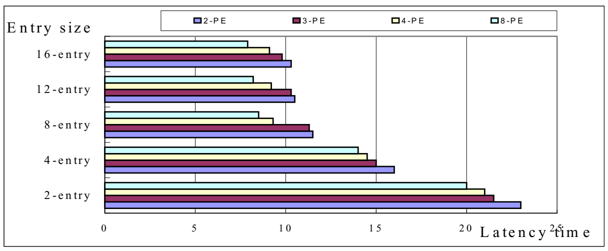


Fig. 7. The communication latency of different request-buffer size.

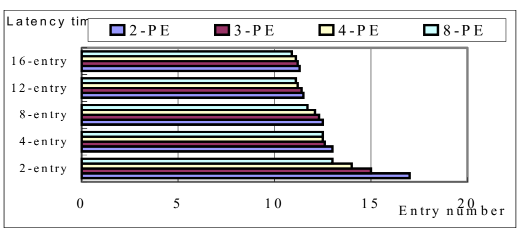


Fig. 8. The Communication latency of different data-queue size.

3.2. The Performance fo Different number of Clients

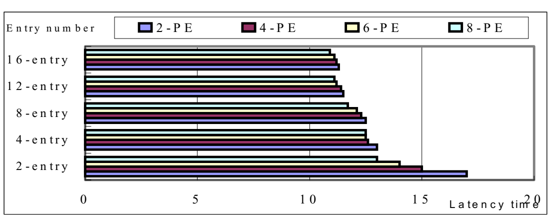


Fig. 9. The Communication latency of different numbers of Processors with the same of buffer sizes under 4-bus.

Figure 9 shows the communication latency of different numbers of processor in case 2, which has the same request-buffer and 4-bus. Referring to Figure 9, we find the lowest communication *latency* occurrence when the number of processor is 8. This reason is caused that the probability of each bus to be used by processors is 0.5. Thus the request commands data of the requesting processor do not expend arbitration time to process, and lower time to process handshaking control among requesting processor, communication unit, and requested processors.

4. Conclusions

This paper we issue the protocol of communication for the discrete computer system. We also propose the algorithms of data receive and data send when this proposed architecture is exploited to the distance learning system. In order to verify our assumption is realizably, we simulate this proposed architecture in a discrete event system and provide it had improved the performance of the DCS.

References

[1] Chih-Yung Chen, Jih-Fu Tu, Jenn Tang, Hsichien Yu, and Dong-Liang Lee, "Exploiting the Request-queue Methodology for Implementing Processor Communication," *Proceedings of the 2003 International Workshop on Distance Education Technology, DET'2003*, Step. 24-26, 2003, Miami, Florida, USA, pp. 814-817.

[2] H. B. Diab and H. S. Tabbars, " The Effect of Inter-PE Communication Network Configuration on the Performance or Array Computer, "*The Proceedings of 25th Annual Simulation Symposium*, 1992, pp. 164-171.

[3] J. L. Hennessy and David Patterson, "Computer Architecture A Quantitative Approach," Morgan Kaufmann Publishers, Inc. Second Edition 1996.

[4] Kai Hwang, "Advanced Computer Architecture Parallelism Scalability Programmability," *McGraw-Hill, Inc.* International Editions 1997.

[5] C. Mung, A. Sutivong, S. Boyd, "Efficient Nonlinear Optimizations of Queuing Systems," IEEE Global Telecommunications Conference, Vol. 3, 17-21 Nov. 2002, pp. 2425-2429.

[6] H.-S. Park, S.-J. Moon, M.-S. Park, B. Kwon, and K.-S. Song, "Design of Inter Processor Communication Controller using ATM Switch and Analysis of its Optimal Message Length Considering Retransmission," *The Proceedings of 4th International Workshop on Real-time Computing System and Applications*, 1997, pp. 18-25.

[7] P. G. Paulin, C. Pilkington, and E. Bensoudane, "Network Processing Challenges and an Experimental NPU [Network Processor Unit]," The Conference and Exhibition of Design, Automation and Test in Europe, 2003, pp. 64-69.

[8] D. Sima, T. Fountain, and P. Kacsuk, "Advanced Computer Architectures: A Design Space Approach," Addison-Wesley, 1995.

A Research in Tennis Teaching E-learning Platform

Shieh-Shing Lin[13], Huay Chang[2], Wen-Chen Tsai[1], Yen-Hao Miao[1]

[1] Department of Electrical Engineering St. John's & St. Mary's Institute of Technology

[2] Department of Information Management, Chihlee Institute of Technology

sslin@mail.sjsmit.edu.tw[3]

ABSTRACT

In this research, we focus on the popular sport--tennis. We present a new teaching method. Base on the e-learning web-site, we design the tennis network teaching. We import expert system to establish the interactive tennis teaching e-learning structure in our research. And we will implement our method in network teaching web-site finally. Our method provides a new teaching model and a new learning channel.

Key words: e-learning, tennis teaching, expert system, interactive method

1. INTRODUCTION

People start concentrating leisure life while their family life becomes steady. Therefore, the outdoor sport exercises become popular gradually. And tennis is one of the most popular sports in our country. But due to the limitations of the tennis knowledge and skill resources in Taiwan, we would like to construct the e-learning platform for tennis teaching. We hope this e-learning program can help the experience exchanging between beginners and trained tennis players. This program would help in breaking out the limitations on teaching program and the promoting on tennis sport.

1.1 Research Motivation

The benefits of the tennis e-learning platform are the resources sharing and the freedom from space limitation. This e-learning platform leads a new channel for those tennis beginners. This is the first motivation of our research.

The interactive learning style is the basic concept of all kinds of learning. The interactive learning strengthens the learning effect and corrects the wrong learning concept immediately. The e-learning model provides the function of interactive learning. It makes the learning process as a communication process and reaches the correct learning goal. This is the second motivation of our research.

While clearing out the barriers of learning process and gathering the people who own the same interesting, these would expands the tennis world. This is the third motivation of our research.

1.2 Research Purpose

The purposes of this research are:

i. We hope to bring the new teaching model and learning channel for the tennis teaching through this research.

ii. We hope to construct the e-learning interactive tennis teaching through this research.

iii. We hope to promote and create a new learning chance of the tennis exercise through this research.

2. LITERATURE REVIEW

In this section, we describe the various knowledge of the tennis and the characteristics of e-learning.

2.1 Tennis

Tennis is invented in the middle of 1400 in France. Tennis is not only a skillful sport but also a sport that requires much knowledge background.

2.1.1 Knowledgeable Tennis

In this section, we have a description in the selection and use of tennis instruments, the competition rules, and the court introduction:

i. The Selection and Use of Tennis Instruments

There are two kinds of tennis: hard-tennis, and soft-tennis. The material of the paddle was wood in the early time. Then, the metal was used in the production of paddle, such as the iron and aluminum. Today, synthetic fiber is widely used.

ii. Competition Rules

The competition rules include scoring method and the difference of hard-tennis and soft-tennis.

(a) Point: The points are accumulated by one point. The rule is: While none of the serve falls into the serve area, the tennist who loses one point. While the game is going on, if the ball was hit outside, the tennis who hit the ball loses point. While the game is going on, if the ball touches the cage, the one who hit the ball loses point, too. While the game is going on, if the competitor couldn't hit back, the competitor loses point. The rules are the same in twosome and foursome competition [1].

(b) Game: Any player has to get four points higher than the competitor, then the player wins a game. The first point is marked '15', the second point is called '30', the third point is marked '50), the four is marked 'game'. 'Game' also means the winning of this game. If the two players got 3 points (40-40), it is called 'deuce'. The point right after the deuce is called 'advantage' that means there is only one point to be the winner. Or it is only a 'deuce'. The word 'love' in tennis means '0',

‘-love’ means ’30-0’.

(c) Set: Any tennist who got six games means that he or she wins one set. If both tennists win five sets, then marked as ‘5 all’. Then, any tennist who would win the set once he or she wins two games sequentially. But if the game is 6:6, then the ‘tie-break’ should be held. If any tennis who wins seven points first and there are two points difference leading the other tennist, the winner is bone. In general, the winner should wins two sets in the competition. But the male grand slam has nine sets competition. The davis cup asks the winner should win the three sets among five sets.

(d) Hard tennis: Twosome, foursome, mix doubles follows the rules: A set has six games. Anyone who leads a set right after he or she wins six games. Soft tennis: A set has seven games. Anyone who leads a set right after he or she wins four games. But the mix doubles should win five games among nine games.

iii. Court Introduction

The different kinds of tennis courts are: a. natural grass tennis court b. en-tout-case c. redbrick sand tennis court d. dust tennis court e. PU plastic tennis court f. asphaltum tennis court g. chamber tennis court. The map of the tennis court is as Fig. 1.

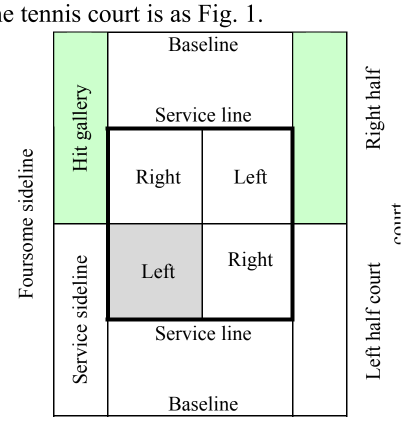


Fig. 1 Tennis court

The side line is 78 inches. The twosome base line is 27 inches. The foursome base line is 36 inches. The middle of the cage is 3 inches, the stob of the side line is 3.5 inches. The distance from base line to serve line is 18 inches. The distance from serve line to the cage is 21 inches. The width of every serve area is 13.5 inches. The width of the gallery is 4.5 inches.

2.1.2 Tennis Skill

The skills of tennis includes:

i. The short method: The description is as follows:

(a) Bump ball: While the ball falls on the ground, then hit it back. (b) Short hit: While the competitor runs to the baseline, hit the ball to the front of the cage that makes the competitor runs hurriedly to catch the ball. (c) Intercept the ball: Intercept the ball that hasn’t fall on the ground [1].

ii. Basic Serve

The serve emphasizes not only the serve point but also the standing position. The first key point is to have the right serve position.

The serve classifies into: The straight serve—that shots backward and appears the fastest speed of serve. The backward shot appears less rotation, fast speed, and hard to control. The rotating serve—that shots athwart from the back. The ball increases slight rotation athwart. This serve is controlled easily and appears fast speed [3]. The straight serve is as Fig. 2.

Fig. 2 The Straight Serve

iii. Forehand & Backhand

Forehand: (a) The preparation actions: Open two feet to the same wide of shoulder while standing. And then, bend down the knee a little, open the elbow a little, hotch forward, and the height of the paddle shouldn’t not be higher than the netter’s eyes. (b) Pull the paddle and turn around: Start roating the right feet while pull the paddle, and hotch the right feet, slightly bending two knees. While pull the paddle, rotate the shoulder, exsert the left hand, keep the balance of the body, relax the elbow that holds the paddle, and fix the radian of the backswing. (c) Step on of the left foot: While backswing and turning around, step on the former (left) foot. Tightly hold the paddle to fix the wrist and keep the paddle the same height of the waist. (d) The shot area: Backswing to the former upper area, and moves athwart to the former foot. The upper shoulder, arm and the waist tightly hold the paddle while the shot moment. The player should lower the jaw, watch the tennis ball concentrate. (e) After shot, the paddle leads the arm based on the tennis ball route. Then, raise the paddle to the show direction. While the paddle stops at the left shoulder, repeat the preparation action. As listed in Fig. 3.

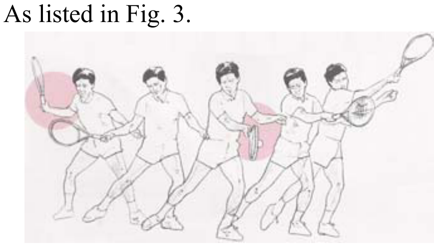


Fig. 3 Forehand

Backhand: (a) Preparation actions: Stand forward, separate two feet to the same wide of shoulder while standing, slightly bend the knees, bend the body forward, move athwart to the toes. Adapt the Eastern Backhand holding method, the other hand hold the middle of the paddle and watch the ball carefully. (b) Backswing and turn around: Start rotating left foot and move athwart to the left foot. Use the other hand that doesn’t hold the paddle to help back-holding the paddle. This keeps the paddle at the serve height. (c)The left foot steps forward: Hold the paddle backward and turning shoulders while needed, and move athwart to the left foot. (d) Shot area: Wave the paddle head slightly, and step at the horizontal direction with the tennis ball's route. Tightly holding the paddle to keep the wrist and make the head of the paddle the same height of the wrist. (e) Hitting zone: Move athwart to the former foot, backswing upward and hold the paddle tightly. The upper shoulder, wrist, arm rotate at the same time, lower the jaw slightly, and watch the tennis ball concentrate. (f) Raise the paddle high and hit the tennis ball. Keep hitting followed the tennis ball's route. (g) Follow through: Extend the arm that holds the paddle at the top of the head, and step forward to face the cage. As listed in Fig. 4.

Fig. 4 Backhand

2.2 e-learning

The e-learning is called network learning, network teaching, on-line learning or electrical learning. E-learning can be classified into synchronous network teaching and asynchronous network teaching. 'Synchronous network teaching emphasizes a real-time information transferring. Through the video meeting and multimedia techniques, the teachers and students can implement asynchronous network teaching at different time and different locations. The teachers can allow the students to join the learning program at anytime through the computer network. The students can submit their questions freely. The students raise their questions through the e-mail, on-line discussion, message board and chat room.

2.2.1 The Characteristics of e-learning

The characteristics of e-learning are: i. Equal educational opportunity. ii. No limitations on time and location. iii. Individual education. iv. Appropriate personal characteristics development. v. The students share the learning with all the students all over the world. vi. Tthe students learn the following abilities: self control, activeness, responsibilities, dependence, and lifelong learning.

2.2.2 The Advantages of e-learning

The e-learning includes the following advantages:

i. Multiple-type of Content: The data transferred through the internet includes: text, graph, image, music, animation and video.

ii. Experience Exchanging Channel: The message board, discussion area and chat room are the communication channels for comment exchanging.

iii. Virtual Learning Groupware: The teaching platform is without the limitation of time and locations, so there are many learners can use the resources at the same time.

iv. Professional Information Providing: The teaching platform specialists provide different topics of teaching materials and comments.

3. RESEARCH METHODOLOGY

The literature analysis method is used by this research. Our teaching platform is designed on the basis of the advantages of present existing e-learning platforms.

3.1 Research Method

First, we collected the literature concerning e-learning and tennis and tried to find out the relationship between e-learning and tennis. We also tried to understanding the construction of e-learning teaching platform.

The sources of literatures contain text books, journals and research reports. We have made a deep straight analysis, cross analysis, and comparison on those literatures. Those analysis results become the base of our research and are widely used on the planning of the e-learning teaching platform.

3.2 Research Steps

The research steps of this research is as Fig. 5.

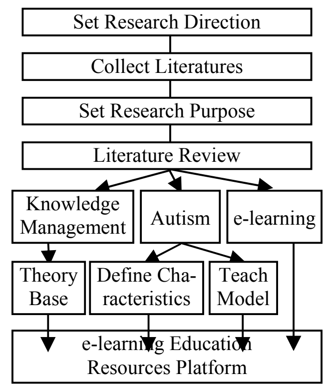


Fig. 5 Research Steps

4. THE PLANNING OF TENNIS TEACHING PLATFORM

In order to create a high-quality e-learning teaching platform, our courses are designed by the following guidelines:

4.1 Meet Learners' Requirements & Centralization Teaching

The courses include different levels. In order to forbid the less attractiveness, the text should be replaced by the video teaching. Those films are produced by the professional specialists. The learners select the appropriate course level to start learning. For those steps those are difficult and unclear, the learners can repeat watching while necessary.

4.2 Interactive Design & Planning to Raise Learning Efficiency

The interactive function is the most important one on the traditional teaching. Therefore, the discussion area, message board, and chat room are installed for interactively communicating on the e-learning teaching platform.

4.3 Uniformity & Simplicity of Learners' Interface

A good e-learning teaching platform should be easily operating and learning by the learners. The interface should be uniformity and simplicity. The learning flows and materials should be understood by the learners conveniently. All those advantages would help in creating the learning strategies for the individual learner.

4.4 Completeness Functions of Teaching Platform

Based on the above three points, the e-learning teaching platform is as Fig. 6.

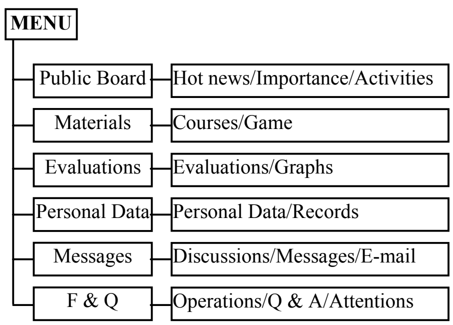


Fig. 6 Functional Structure

i. Public Board: This public board is to offer public messages, important issues, activities news, information updating. The fans of tennis can hold up the hot news.

ii. Courses & Materials: The main subject of the courses is the skill teaching. It uses the characteristics of the multimedia and combines with the introduction of animation and text to increase the efficiency of the skill.

iii. Teaching Evaluation: The learners can use the questions that are listed on the e-learning teaching platform to evaluate their learning efficient.

iv. Personal Data: This e-learning teaching platforms keeps the entire learning records those can be used to arrange better connection for the next learning stage.

v. Experience Exchange: The discussion area provides a ground for the learners sharing the learning experience. These channels act like a two-way communication channel for the learners and professional coaches.

vi. F&Q: This part includes the operating method, use instructions and limitation. This part leads the learners to be easily familiar with the platform.

5. CONCLUSION

Through the construction of the e-learning teaching platform, our research provides the positive advantages of the tennis teaching. The advantages are:

5.1 The Convenience of Learning Channel

The platform is set on the internet. Once the personal computer and network communication equipments are prepared, we can access to the platform and retrieve/store the materials. According to the personal status, the suitable courses can be selected individually. The materials arranged by the platform are professional and practical for the tennis learners. All the learners can also share their learning and exercising experience through the platform.

5.2 The Multiple-types of Learning Content

The learning objective of tennis includes the selection of tennis tools and the learning of skills. The computer can represent the text, pictures, audio, animation and video. Therefore, if there are complicate and serial movements of the tennis, the computer would give a great help. The learners can watch the film repeatedly. And the film can be paused at any time.

5.3 The Gathering of Network Group

In tennis world, the coaches and the well-experienced tennis players own much experience. The professional skills of the coaches and the well-experience could share with all the people who access to the platform. The ultimately resources create the network groups through the network.

5.4 The Immediate Delivery of Message

For those people who access to the platform play as a message sender and a message receiver. Everyone can send the message through the network and that makes the message transferring rapidly. The function of the speed raises the attraction of the tennis players.

5.5 The Endless Limit of Learning Policy

The advantage of the e-learning platform is that there is no limitation on time and locations. Therefore, the learners can start learning depending on his or her personal qualifications.

Once the learners are learning through the e-learning teaching platform, if there are any problems, the learners can raise the questions at anytime. Or the learners can watch the film repeatedly. The teachers can evaluate the learners based on the access data.

REFERENCE

[1] Nei, Shan-Sheng, The Tennis, Taipei, 1988.

[2] Wang, Shi-Sheng and Jan-Jywe Liu, Taipei.

[3] Bei-Shr, Jen-Re, The Basic Tennis, Taipei, 2000.

[4] Ma, Da-Di, The Tennis Skills, Taipei, 2002.

[5] Chang, Jya-Yi, De-Sheng Hwung, 'The Long Distance Education and Future Education', Journal of Information and Education, vol.90.

[6] Chen, Yun-Long and Yi-Chan Jwung, 'The Knowledge Value Chain', Taipei, China Productivity Center, 2003.

[7] Wu, Tyan-Fang, Shywe-Lyang Fan and Bo-Shen Chang, 'The Strategy of e-learning Construction', Journal of Education Information & Research, vol.54, pp.88-89, 2003.

[8] Hwung, R.H., 'The e-learning Strategy in Campus', Journal of Teachers, voo.179, pp.26-28, 2004.

[9] Wu, Hwung-Rung, The Tennis, Taipei, 2004.

[10] Tennis, Norya Jungle Publications, Taipei, 2004.

[11] Tsung, Chyang, The Tennis Skills, Taipei, 2005.

[12] Tennis Teaching http://www.ntut.edu.tw/~wwwtennis/Untitled7.htm

[13] International Tennis Story http://tennis.sportsmag.com.tw/

[14] ROC Tennis Association http://www.tennis.org.tw/

Rule-based Knowledge Similarity Distance Calculation System-Using Fuzzy Conditional Probability Knowledge Similarity Algorithm

Chin-Jung Huang[1], Ying-Hong Lin[2]

[1]Department of Mechanical Engineering
St. John's & St. Mary's Institute of Technology

[2]Department of Information Management
St. John's & St. Mary's Institute of Technology

Abstract—In **the processes of knowledge accumulate, due to knowledge's source from different expert, this way cause logical or structure errors. Because time and space environment change so that appear as new technologies, new laws, new methods and new evidences, so knowledge may not be suitable for use now. Meanwhile application of the wrong knowledge causes wrong decisions, for the knowledge base, it is the correctness of the knowledge that is most important. But seldom discusses how to add value to the knowledge, and add value to the knowledge before we want to confirm knowledge's relationship among, so the similarity calculate of knowledge is very important.**

This study integrates Fuzzy Theory, Conditional Probability, Vector Matrices and Artificial Intelligence, to establish the Fuzzy Conditional Probability Knowledge Similarity Algorithm (FCPKSA) and develop the Rule-based Knowledge Similarity Distance Calculation System (RKSDCS). This system can rapidly obtain knowledge similarity matrix.

Keywords: Rule-Based Knowledge, Fuzzy Theory, Conditional Probability, Vector Matrices, Artificial Intelligence, Similarity

1. Introduction

In the processes of knowledge accumulation, the knowledge sources are derived from different experts, this way cause logical or structure errors. Because time and space environment change so that appear as new technologies, new laws, new methods and new evidences, so knowledge may not be suitable for use now, and common issues may include existing redundancy rules, conflict rules, circularity rules, and incompleteness rules. Because an application of wrong knowledge leads to wrong decisions, for the knowledge base, it is the correctness of knowledge that is most important. The quality of knowledge overweighs its quantity. This is especially applicable to the current application of knowledge focusing on its Discovery, Creation, Preservation, Delivery, Sharing and Directly Use. It seldom discusses how to add value to the knowledge. For example, the accuracy of knowledge requires checking, modifying or deleting, complete similar knowledge to do merging or integration, with conflict or overlapping knowledge to do modifications or updates, and innovative inference of new knowledge as an added value treat. But add value to the knowledge before we want to confirm knowledge's relationship among, so the similarity calculation of both knowledge is very important.

Based on the causes above, this study proposes the Fuzzy Conditional Probability Knowledge Similarity Algorithm (FCPKSA) and develops the Rule-based Knowledge Similarity Distance Calculation System (RKSDCS). This system can rapidly obtain knowledge similarity matrix.

2. Review of Related Research

To establish rule-based expert system emphasizing in inference rules whether they are existing logical errors or structural errors. In other word, we mainly discuss verification and validation of inference rules in expert system, This is seldom discussed to upgrade knowledge's application added value or knowledge reuse [1] [2] [3] [4]. 2001 M.LYNNE MARKUS proposed " Toward a Theory of Knowledge Reuse: Types of Knowledge Reuse Situations and Factors in Reuse Success" [5], to discuss success factors of knowledge reuse.

3. The Similarity Calculation of Rule-based Knowledge

3.1 Knowledge Representation

The syntax of rule-based knowledge representation was IF <antecedent> THEN <consequent>. The antecedent or consequent would be expressed as a sentence. This study uses the following four components: Object (O), Attribute (A), Relationship Operator (R), and Linguistic Value (V) to combine the O-A-RV format structure as Fig. 1, for example the O-A-RV format structure of a sentence is represented as Table 1.

Any attribute of objects in knowledge through proper transform mapping to number data will be able to use the three components of O-A-RV to express the antecedent vector and the consequent vector. If the antecedent has n attributes, it may use 3*n components to express itself as in eq. (1). The consequent vector according to express way of antecedent vector before, meanwhile antecedent vector and consequent vector combine to form the knowledge vector.

3.2 Transform Mapping of O-A-RV

3.2.1 Transform Mapping of the O-A part

Since the data style of the O-A part is a string in a knowledge sentence, then O-A can be individually expressed according to whether they can be assigned the same characters to transform mapping to 0 or 1 as shown Table 2.

Table 2 Transform Mapping of the O-A part

O	A	O- A Mapping value	
Exist be assigned same character		1	1
Not exist be assigned same character		0	0

3.2.2 Transform Mapping of the RV Part

The data style of V in a sentence may appear as shown in Table 3.If the data style of V is interval and ratio, then V retain it's original value; if the data style of V is ordinal, then V transforms mapping by Table 4 or use Membership Function as shown in Fig. 2 to Fig. 3; if the data style of V is nominal, then V according to weather exist be assigned the same character to transform mapping to 0 or 1. If the data style of V is a number and R is an equal's symbol in whole knowledge, then use eq. (2) to normalize it and allows its value is between 0 and 1.

$$V_{norm} = \frac{V - V_{\min}}{V_{\max} - V_{\min}} \qquad (2)$$

V_{norm}: V normalizated value, the value is between 0 and 1

V: Linguistic Value V.

V_{max}: Maximum of V in knowledge cases.

V_{min}: Minimum of V in knowledge cases.

Table 3 Data Type of Linguistic Value (V)

Type	Property of Operator
Nominal	Unable to compare its magnitude. Unable to do arithmetic calculations.
Ordinal	Its quantity is finite, and its with an order relationship. It's possible to compare its magnitude, but not possible to do arithmetic calculations.
Interval and ratio	Its data style is numerical.

Table 4 Mapping of Ordinal Data

Description of V	Mapping Value	Remark
Very Small	0	V directly is assigned the suitable mapping value between 0 and 1 or use the Membership Function of the Fuzzy Theory to do transform mapping.
Small	0.25	
Medium	0.5	
Large	0.75	
Very Large	1	

If the data style of V is a number and R is not an equal's symbol, then transform mapping of RV may use conditional probability $P(B \mid A) = \frac{P(B \cap A)}{P(A)}$ to calculate, and its mapping value is between 0 and 1. If the mapping value is greater than 1, then take it's value is recorded as 1. Here, Here, P(A) is the probability of event A happening, and $P(A \cap B)$ is the probability of event A and event B occurring at same time.

Assume x expresses the value of Testing Cases (T), and y express the value of Knowledge Cases (K), then transform mapping of RV in different situations as shown in Table 5 to Table 6. But the transform mapping of RV in other situations can not listed in paper for pages limit.

Term V_1 is the relationship of x and y, V_2 is the relationship of K and y, and V_3 is the relationship of T and x.

max: the maximum of V of Knowledge Cases.

min: the minimum of V of Knowledge Cases.

Table 5 Transform Mapping of RV for V_1:x > y

V_3 \ Value V_2	K > y	K = y	K < y
T > x	$\frac{\max - x}{\max - y}$	0	0
T = x	$\frac{1}{\max - y}$	0	0
T < x	$\frac{x - y}{x - \min}$	1	1

Table 6 Transform Mapping of RV for V_1:x < y

V_3 \ Value V_2	K > y	K = y	K < y
T > x	1	1	$\frac{y - x}{y - \min}$
T = x	0	0	$\frac{1}{y - \min}$
T < x	0	0	$\frac{x - \min}{y - \min}$

3.3 Fuzzy Conditional Probability Knowledge Similarity Algorithm

3.3.1 Knowledge Similarity Calculation

When the antecedent and the consequent have finished transform mapping and is normalized, then the knowledge can be expressed by the O-A-RV format and the composed knowledge vector. The sentence of the antecedent or consequent may also be able to use the connective operator {"AND", "OR"} to combine.

If the quantity of knowledge is m, the antecedent is with n dimensions and the consequent is with l dimensions, then the Knowledge Matrix (K), Antecedent Matrix (A), and Consequent Matrix (C) is defined as shown in eq. (3):

$$K = [k_{ij}]_{m*(n+l)} = \begin{bmatrix} k_{11} & k_{12} & \cdots & \cdots & k_{1(n+l)} \\ k_{21} & k_{22} & \cdots & \cdots & k_{2(n+l)} \\ \vdots & \vdots & & & \\ \vdots & \vdots & & & \\ k_{m1} & k_{m2} & \cdots & \cdots & k_{m(n+l)} \end{bmatrix} \quad (3)$$

$$= \begin{array}{cc} \text{Sentence} & \text{Consequent Matrix (C)} \\ \multicolumn{2}{c}{\left[\begin{array}{ccccc:ccccc} a_{11} & a_{12} & \cdots & \cdots & a_{1n} & c_{11} & c_{12} & \cdots & \cdots & c_{1l} \\ a_{21} & a_{22} & \cdots & \cdots & a_{2n} & c_{21} & c_{22} & \cdots & \cdots & c_{2l} \\ \vdots & \vdots & & & & \vdots & \vdots & & & \\ \vdots & \vdots & & & & \vdots & \vdots & & & \\ a_{m1} & a_{m2} & \cdots & \cdots & a_{mn} & c_{m1} & c_{m2} & \cdots & \cdots & c_{ml} \end{array}\right]} \end{array}$$

When any knowledge is expressed as a knowledge vector, for example $\vec{k}_i = (k_{i1}, k_{i2}, k_{i3}, \cdots; k_{i(n+l)})$, $\vec{k}_j = (k_{j1}, k_{j2}, k_{j3}, \cdots; k_{j(n+l)})$ are both knowledge vectors, this study proposes that the product of Distance Similarity (DS) and Angle Similarity (θS) to represent the both knowledge similarity (KKS) as eq. (4). The calculation of DS uses Euclidean Distance to avoid the similarity of the maximum distance knowledge as zero as shown in eq. (5) to eq. (6); the calculation of θS uses the Inner Product as eq. (7). By the same way token the similarity of both antecedents (AAS) and both consequents (CCS), the similarity of antecedents and consequents (ACS) can act individual as shown in eq. (8) to eq. (10).

$$\text{KKS}_{ij} = DS(\vec{k}_i, \vec{k}_j) \cdot \theta S(\vec{k}_i, \vec{k}_j) \quad (4)$$

In eq. (4), the values of DS and θS are between 0 and 1.

The Euclidean Distance is defined to be $D(\vec{k}_i, \vec{k}_j) = \|\vec{k}_i - \vec{k}_j\| = \sqrt{\sum_{v=1}^{n+l} |\vec{k}_{iv} - \vec{k}_{jv}|^2}$, the Length is defined to be $\|\vec{k}_i\| = \sqrt{\sum_{v=1}^{n+l} |k_{iv}|^2}$, and the Inner Product is defined to be $< \vec{k}_i, \vec{k}_j >= \sum_{v=1}^{l+n} k_{iv} k_{jv}$, then

$$DS(\vec{k}_i, \vec{k}_j) = [1 - (\frac{D(\vec{k}_i, \vec{k}_j)}{\alpha \cdot \max_{\forall i,j} D(\vec{k}_i, \vec{k}_j)})] \quad (5)$$

$$\text{The constant coefficient } \alpha = \frac{1}{1 - DS_{\min}} \quad (6)$$

And α >1, where DS_{min} is the minimum of in whole DS, but DS_{min} >0, and maximum of DS_{min} may be decided by the user. For example when DS_{min}=0.1 then α=1.11.

$$\theta S(\vec{k}_i, \vec{k}_j) = \frac{< \vec{k}_i, \vec{k}_j >}{\|\vec{k}_i\| \|\vec{k}_j\|} = \cos\theta \quad (7)$$

$$\text{AAS}_{ij} = DS(\vec{a}_i, \vec{a}_j) \cdot \theta S(\vec{a}_i, \vec{a}_j) \quad (8)$$

$$\text{CCS}_{ij} = DS(\vec{c}_i, \vec{c}_j) \cdot \theta S(\vec{c}_i, \vec{c}_j) \quad (9)$$

$$\text{ACS}_{ij} = DS(\vec{a}_i, \vec{c}_j) \cdot \theta S(\vec{a}_i, \vec{c}_j) \quad (10)$$

When the KS of both knowledge is higher, this expresses that both knowledge is more similar, the KKS is between 0 and 1.When KKS=1, this expresses that both knowledge is completely the same, when KKS=0, this expresses that both knowledge is complete not the same. When a vector is Zero Vector, then its corresponding similarity is zero.

3.3.2 Similarity Matrix

The definitions of Knowledge, Antecedent, Consequent, Antecedent and Consequent Similarity Matrix are displayed as eq. (11) to eq. (14).

(1) Knowledge Similarity Matrix (KSM):

$KSM=[KKS_{ij}]_{m*(n+l)}$, and $KKS_{ij}=1$, when $i=j$ (11)

Subject to $i=1$ to m, $j=1$ to $(l+m)$

(2) Antecedent Similarity Matrix (ASM)：

$ASM=[AAS_{ij}]_{m*n}$, and $AAS_{ij}=1$, when $i=j$ (12)

Subject to $i=1$ to m, $j=1$ to n

(3) Consequent Similarity Matrix (CSM)：

$CSM=[CCS_{ij}]_{m*l}$, and $CCS_{ij}=1$, when $i=j$ (13)

Subject to $i=1$ to m, $j=1$ to l

(4) Antecedent Consequent Similarity Matrix (ACSM)：

$ACSM=[ACS_{ij}]_{n*l}$, (14)

Subject to $i=1$ to n, $j=1$ to l

3.3.3 Fuzzy Conditional Probability Knowledge Similarity Algorithm

According to the analysis above, this paper integrates Fuzzy Theory, Conditional Probability, Vector Matrices, and Artificial Intelligence Rule Based Inference to construct the Conditional Probability Knowledge Similarity Algorithm (FCPKSA) and develop a Rule-based Knowledge Similarity Distance Calculation System (RKSDCS). Fig 4 is the architecture of FCPKSA, the flow chart of FCPKSA as shown in Fig. 5.

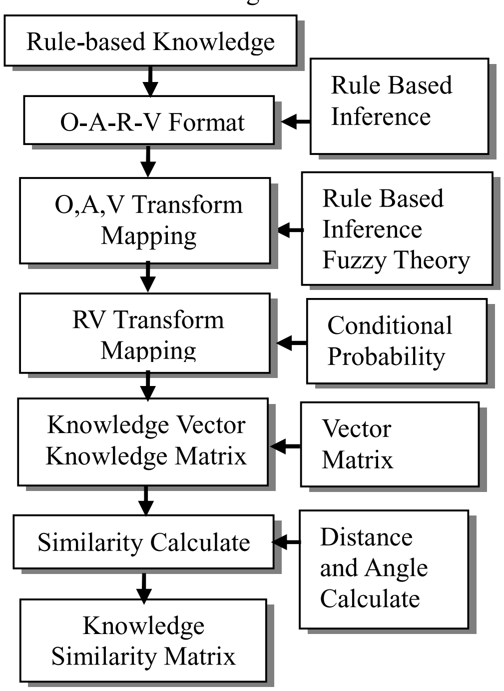


Fig. 5 Flow Chart of FCPKSA

3.3.4. Knowledge Similarity Calculation of Instances

The rule-based knowledge instances are expressed as shown Fig. 8. Within Fig. 8 we have fourteen cases, which are expressed to the knowledge matrix by the O–A-RV format. If the knowledge components have different dimensions, then all elements of insufficient attributes take as zero to calculate. Any pairwise knowledge utilizes eq. (5) to eq. (9) to proceed with one hundred sixty-nine times pairwise mapping calculations. The ASM, CSM, and KSM of knowledge instances are expressed as shown Fig. 9 to Fig. 11.

3.4 The Implementation of RKSDCS

According to the FCPKSA, we developed the web page of Rule-based Knowledge Similarity Distance Calculation System (RKSDCS). The home page address of the system is http://kamd.dnip.net/RBKSCS/. It can quickly and effectively outputs the KKM, AAM, and CCM. It is able to apply at distance education of knowledge management course, too.

Fig. 6 is the main page of the system. Fig. 7 is import the Microsoft Excel knowledge data file. Fig.8 is the knowledge instances. The output result of similarity matrices are as shown Fig. 9 to Fig. 11, and the V transform mapping use Membership Function in Fig 2 and Fig 3.

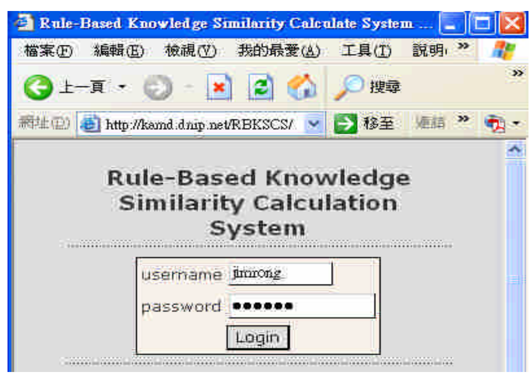


Fig. 6 The main page of RKSDCS

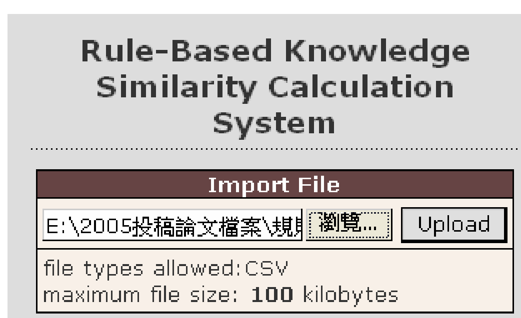


Fig. 7 Imported the Knowledge File

4 Conclusions and Future Work

According to the analysis above, we conclude the following results:

(1) This study propose to use Object, Attribute, Relationship Operator, and Linguistic Value four components, which express knowledge vectors by the O–A-RV format.

(2) This study propose to use distance and angle calculate similarity method.

(3)We integrated Conditional Probability, Vector Matrices and Artificial Intelligence, to establish the Fuzzy Conditional Probability Knowledge Similarity Algorithm (FCPKSA) and develop the Conditional Probability Knowledge Similarity Distance Calculate System (CPKSDCS), can rapidly to obtain knowledge's similarity among to build knowledge similarity matrix. It is able to apply at distance education of knowledge management course, too.

(4)The paper builds six knowledge's relationships, and let knowledge do added value treatment and upgrade knowledge's application added value.

Future work of this study is listed as following:

(1)When the data style of V is ordinal, then V use the Membership Function of Fuzzy Theory to transform mapping.

(2)According to CPKSDCS, we can rapidly obtain the most similar knowledge case subject to the testing case, the application of Case-Based Reasoning (CBR) will be to predict or assist decision.

References

[1]Seng-Ming Liu, "A Novel Approach for Verifying Fuzzy Knowledge-Based Systems", Master thesis, Department of Information Management at National Chi Nan University, Taiwan, 2004

[2] J. R. Geissman and R. D. Schultz, "Verification and Validation of Expert Systems", AIExpert, Vol.3, No.2, pp.26-33, 1988.

[3] Wei-Tek Tsai, Rarna Vishnuvajjala and Du Zhang, "Verification and Validation of Knowledge-Based Systems", IEEE Transactions on Knowledge and Data Engineering, Vol. 11, No. 1, pp. 202-212 , 1999

[4] D. Zhang and D. Nguyen "PER_PARE: A Tool for Knowledge Base Verification," IEEE Transactions on Knowledge and Data Engineering, Vol. 6, No.6, pp.983-989,1994.

[5]M.LYNNE MARKUS, "Toward a Theory of Knowledge Reuse: Types of Knowledge Reuse Situations and Factors in Reuse Success", Journal of Management Information Systems, Vol. 18,No. 1 pp.57-93,2001

Sentence =[Object] + [Attribute] + [Relationship Operator] + [Linguistic Value]

Fig. 1 Components of Sentence

Table 1 O-A-RV Format Representation of a Sentence

Sentence	O	A	R	V
The temperature of the engine is more than 100 °c.	Engine	temperature	>	100 °c
The vehicle's color is red.	Vehicle	color	=	red
The span of Bridge is less than 50m.	Bridge	span	<	50m

$$\text{Antecedent vector} =[\underbrace{O_1\ \ A_1\ \ R_1V_1}_{1^{st}\text{ Attribute}}\ \ \underbrace{O_2\ \ A_2\ \ R_2V_2}_{2^{nd}\text{ Attribute}}\ \ \underbrace{O_3\ \ A_3\ \ R_3V_3}_{3^{th}\text{ Attribute}}\ldots\ldots\underbrace{O_n\ \ A_n\ \ R_nV_n}_{n^{th}\text{ Attribute}}] \quad (1)$$

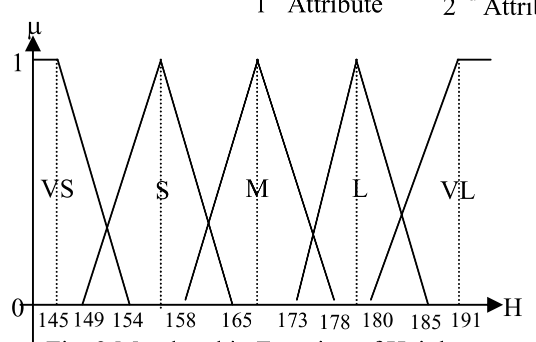


Fig. 2 Membership Function of Height

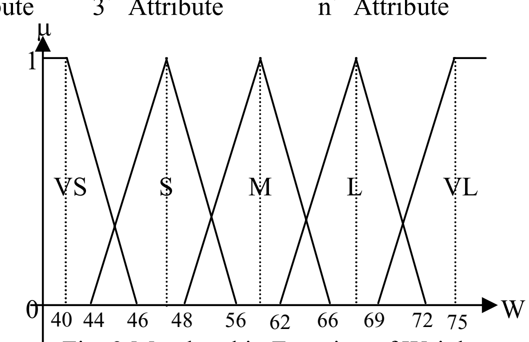


Fig. 3 Membership Function of Weight

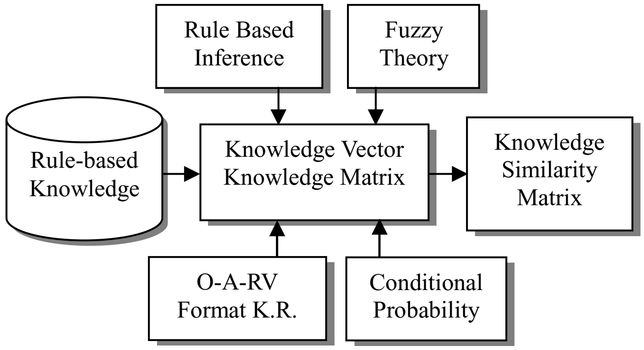


Fig. 4 Architecture of FCPKSA

Rule-Based Knowledge Similarity Calculation System

Table.1 Knowledge Instances

Antecedents					Consequents				
No.	O	A	R	V	No.	O	A	R	V
A1	M	Height	>	190	C1	M	Height	=	Very_large
A2	M	Weight	=	45	C2	M	Weight	=	Very_small
A3	M	Weight	<	55	C3	M	Weight	=	small
A4	F	Height	>	180	C4	F	Height	=	Very_large
A5	F	Weight	>	70	C5	F	Weight	=	Very_large
A6	M	Height	=	173	C6	M	Height	=	medium
A7	F	Weight	=	48	C7	F	Weight	=	medium
A8	M	Weight	=	65	C8	M	Weight	=	medium
A9	M	Height	=	195	C9	M	Height	=	Very_large
A10	F	Height	=	160	C10	F	Height	=	medium
A11	F	Weight	<	40	C11	F	Weight	=	Very_small
A12	F	Height	<	150	C12	F	Height	=	small
A13	M	Height	>	180	C13	M	Height	=	large
A14	M	Height	<	162	C14	M	Height	=	small

Fig. 8 Knowledge Instances

	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14
A1	1.00	0.11	0.11	0.22	0.01	0.35	0.00	0.11	1.00	0.11	0.00	0.11	0.57	0.35
A2	0.11	1.00	0.36	0.00	0.11	0.11	0.11	0.35	0.11	0.00	0.11	0.00	0.11	0.11
A3	0.11	1.00	1.00	0.00	0.11	0.11	0.35	0.35	0.11	0.00	0.35	0.06	0.11	0.23
A4	0.35	0.00	0.00	1.00	0.15	0.11	0.11	0.00	0.35	0.35	0.11	0.35	0.35	0.11
A5	0.11	0.11	0.11	0.35	1.00	0.11	0.35	0.11	0.11	0.35	0.35	0.28	0.11	0.08
A6	0.35	0.11	0.11	0.11	0.00	1.00	0.00	0.11	0.35	0.11	0.00	0.11	0.35	0.35
A7	0.00	0.11	0.11	0.11	0.35	0.00	1.00	0.11	0.00	0.11	0.35	0.11	0.00	0.00
A8	0.11	0.35	0.35	0.00	0.11	0.11	0.11	1.00	0.11	0.00	0.11	0.00	0.11	0.11
A9	0.48	0.11	0.11	0.13	0.00	0.35	0.00	0.11	1.00	0.11	0.00	0.11	0.39	0.35
A10	0.11	0.00	0.00	0.35	0.11	0.11	0.11	0.00	0.11	1.00	0.11	0.35	0.11	0.11
A11	0.00	0.11	0.32	0.11	0.35	0.00	0.35	0.11	0.00	0.11	1.00	0.20	0.00	0.04
A12	0.11	0.11	0.11	0.35	0.28	0.11	0.35	0.11	0.11	0.35	0.35	1.00	0.11	0.34
A13	1.00	0.11	0.11	0.35	0.02	0.35	0.00	0.11	1.00	0.11	0.00	0.11	1.00	0.35
A14	0.35	0.35	0.35	0.11	0.08	0.35	0.11	0.35	0.35	0.35	0.11	0.35	0.35	1.00

Fig. 9 ASM of Knowledge Instances (Subject to DS_{min}=0)

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14
C1	1.00	0.17	0.25	0.11	0.03	0.72	0.00	0.15	1.00	0.17	0.11	0.26	0.90	0.60
C2	0.14	1.00	0.84	0.00	0.30	0.26	0.29	0.97	0.14	0.02	0.14	0.02	0.19	0.29
C3	0.23	0.84	1.00	0.00	0.29	0.31	0.25	0.80	0.23	0.05	0.23	0.05	0.27	0.31
C4	0.09	0.13	0.10	1.00	0.34	0.33	0.30	0.13	0.18	0.97	0.18	0.87	0.25	0.35
C5	0.01	0.30	0.28	0.29	1.00	0.02	0.90	0.30	0.01	0.30	0.46	0.28	0.01	0.02
C6	0.72	0.27	0.31	0.22	0.06	1.00	0.00	0.26	0.72	0.27	0.05	0.31	0.83	0.89
C7	0.00	0.29	0.24	0.30	0.90	0.00	1.00	0.29	0.00	0.29	0.35	0.24	0.00	0.00
C8	0.13	0.97	0.80	0.00	0.30	0.24	0.30	1.00	0.13	0.01	0.13	0.01	0.17	0.28
C9	1.00	0.17	0.25	0.11	0.03	0.72	0.00	0.15	1.00	0.17	0.11	0.26	0.90	0.60
C10	0.15	0.02	0.02	0.89	0.30	0.26	0.29	0.02	0.15	1.00	0.15	0.84	0.19	0.29
C11	0.11	0.17	0.25	0.11	0.46	0.08	0.35	0.15	0.11	0.17	1.00	0.26	0.10	0.06
C12	0.23	0.05	0.06	0.71	0.29	0.31	0.25	0.05	0.23	0.84	0.23	1.00	0.27	0.31
C13	0.90	0.21	0.29	0.15	0.05	0.83	0.00	0.19	0.90	0.21	0.09	0.29	1.00	0.71
C14	0.60	0.29	0.31	0.26	0.04	0.89	0.00	0.29	0.60	0.30	0.03	0.31	0.71	1.00

Fig. 10 CSM of Knowledge Instances (Subject to DS_{min}=0)

	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	K13	K14
K1	1.00	0.13	0.17	0.16	0.02	0.50	0.00	0.12	1.00	0.14	0.04	0.17	0.69	0.46
K2	0.12	1.00	0.52	0.00	0.19	0.17	0.18	0.53	0.12	0.01	0.12	0.01	0.14	0.18
K3	0.16	0.89	1.00	0.00	0.18	0.19	0.30	0.51	0.16	0.02	0.28	0.06	0.17	0.26
K4	0.20	0.00	0.00	1.00	0.21	0.15	0.19	0.00	0.20	0.52	0.09	0.48	0.23	0.17
K5	0.04	0.19	0.18	0.32	1.00	0.07	0.52	0.19	0.04	0.33	0.39	0.28	0.05	0.07
K6	0.49	0.18	0.19	0.16	0.02	1.00	0.00	0.17	0.49	0.16	0.02	0.19	0.49	0.53
K7	0.00	0.18	0.17	0.19	0.52	0.00	1.00	0.18	0.00	0.19	0.34	0.16	0.00	0.00
K8	0.11	0.53	0.51	0.00	0.19	0.16	0.18	1.00	0.11	0.00	0.11	0.00	0.11	0.18
K9	0.64	0.13	0.17	0.12	0.01	0.50	0.00	0.12	1.00	0.11	0.04	0.17	0.57	0.46
K10	0.09	0.00	0.00	0.53	0.18	0.15	0.19	0.00	0.09	1.00	0.09	0.48	0.09	0.17
K11	0.04	0.13	0.29	0.11	0.40	0.03	0.35	0.12	0.04	0.11	1.00	0.23	0.04	0.05
K12	0.16	0.08	0.09	0.49	0.29	0.19	0.30	0.08	0.16	0.49	0.29	1.00	0.16	0.33
K13	0.93	0.15	0.18	0.24	0.04	0.52	0.00	0.14	0.93	0.15	0.03	0.18	1.00	0.49
K14	0.45	0.32	0.33	0.17	0.08	0.52	0.07	0.32	0.45	0.31	0.06	0.33	0.45	1.00

Fig. 11 KSM of Knowledge Instances (Subject to DS_{min}=0)

The Experiences for Implementation the Location Based Tour Guide System

Jenn Tang P.-K. Wang[*] S.-Y. Tsao[**]

Department of Information Management
National Taipei College of Business, Hwa Hsia Institue of Technology[*] , Chihlee Institute of Technology[**] , Taiwan
bestmails@gmail.com wangpk@cc.hwh.edu.tw[*] sytsao@mail.chihlee.edu.tw[**]

Abstract

This system will, based on IEEE 802.11 series, implement a wireless location base Tour Guide service system suitable for "indoor", "short-range" and "small-scale" applications. Through the characteristics of path loss specific to the electric wave of Access Point (AP), this system can capture the location of users via three-point positioning mode. When the users hold the mobile devices, this system will automatically display the location of users in order to provide diversified real-time information services in a flexible and rapid manner. The following features could be found from the experiences of this system: (1) acceptable accuracy: in order to control the latest position of users, the accuracy of a permissible allowance should be further enhanced within an indoor positioning environment; (2) expanded information: more diversified value-added services and applications can be expanded in conjunction with "electronic map"; (3) automatic tour guide service: take tour guide service for example, the location information of users can be obtained through "wireless positioning system," which can actively or passively transmit location-related tour guide information to the users for an automatic tour guide.

***Keywords*: tour guide, wireless, location base.**

1. Introduction

1.1 Research Motivations

The traditional wireless location based service (LBS) system generally comprises a Global Positioning System (GPS) and a Global System for Mobile Communication (GSM) as well as infrared technology, etc. GPS and GSM, designed for outdoor applications, are not suitable for the development of this system. As the wireless network-based campus tour guide system is only required to position indoor areas of departments/divisions, the extremely precise infrared positioning mode is not well-suited for this purpose. So, this system attaches great importance to the positioning technology based on the research of WLAN indoor positioning technology, and implements the system by combining "Multiple Nearest Neighbors" positioning method of three-point positioning mode. Without additional hardware setting, WLAN requires only mobile devices for use with wireless network card and Access Point (AP), whereby the web page can display the positioning services rendered by this system.

This research is intended to (1) develop a system to offer accurate positioning services; (2) expand the development of application technologies for wireless network positioning; (3) use the mobile devices to meet the increasing demands of various users; (4) push forward the trend of commercial and electronic application services for increased commercial benefits. Coupling with the wide applications of mobile devices and WLAN, there is a growing trend for the demanding indoor or outdoor positioning services. Given that indoor positioning service is based on the possibility of getting an easy and automatic access to the positioning information, this research seeks to explore how to develop an efficient positioning system and automatically provide the users with accurate service information.

1.2 Research Purposes

We have the following research purposes:

(1) For "service position server", the associated services such as "electronic map" can be designed for the users within a positioning range of "wireless positioning system".

(2) Take "campus tour guide of Chihlee Institute of Technology" for example, "Service Position Server (SPS)" can actively or passively transmit location-related services to the users.

2. Literature Review

2.1 IEEE802.11

2.1.1 Definition

The Institute of Electrical and Electronics Engineers (IEEE) set up 802.11 workgroup in 1990. IEEE standard was initiated in 1997 to become a key standard of WLAN, with its major purpose of formulating the uniform specifications of WLAN in Medium Access Control (MAC) and Physical Layer (PHY) of OSIRM model, which is referred to as "IEEE Standard".

2.1.2 Extended Service Set of IEEE 802.11

The wireless workstations and access points are established via two 802.11-compliant modes:infrastructure mode and ad hoc mode. In the infrastructure mode, all workstations within this system are linked to single access point, rather than interconnection between workstations. In the ad hoc mode, the workstations enable direct telecommunication without the assistance of access points. (Swaminatha and Charles, 2003) As shown in Fig. 2-1, the infrastructure mode comprises access points and workstations under the same radio coverage of Basic Service Set (BSS). The interconnection mode of BSS is conditioned upon the demands of users, models of workstations or instruments and commercial considerations, etc. (Swaminatha and Charles, 2003)

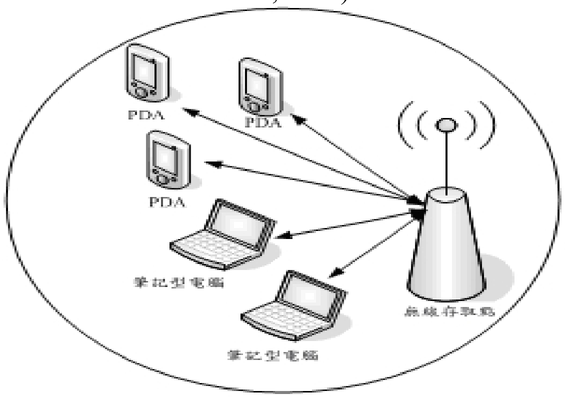


Fig. 2-1 Basic Service Set, BSS

Under the framework of ad hoc mode, the workstations enable telecommunication without the assistance of access points (refer to Fig. 2-2). As this on-the-fly mode is not required to be linked to the wired network, and easy to be integrated and decomposed, every node can communicate directly with other nodes. But, if it is planned to link to other wired or wireless network in ad hoc mode, some restrictions shall apply owing to the fact that every node is not available with a master-slave relation, or the workstations shall maintain the independency. Like the infrastructure mode, the interconnection of several workstations constitutes BSS. (Swaminatha and Charles, 2003)

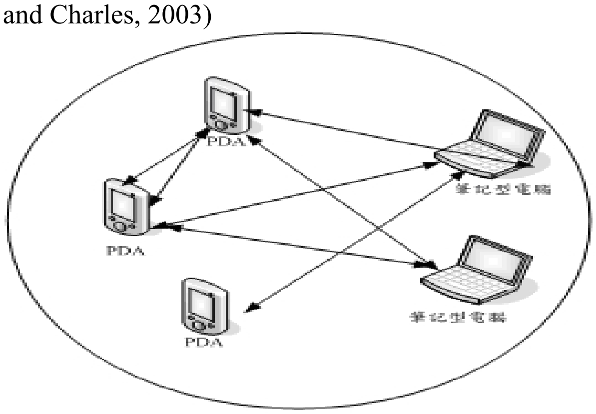


Fig. 2-2 BSS, Ad Hoc

The interconnection of some BSSs will form a distributed system, whereby it is possible to set up a large-scale network framework for an extended wireless coverage. Such distributed system is called Extended Service Set (ESS) (Swaminatha and Charles, 2003). Owing to limited service range of BSS, 802.11 allows to link some BSSs for an extended service set (ESS), thereby expanding the coverage of wireless network. Fig. 2-3 shows a set composed of several BSSs.

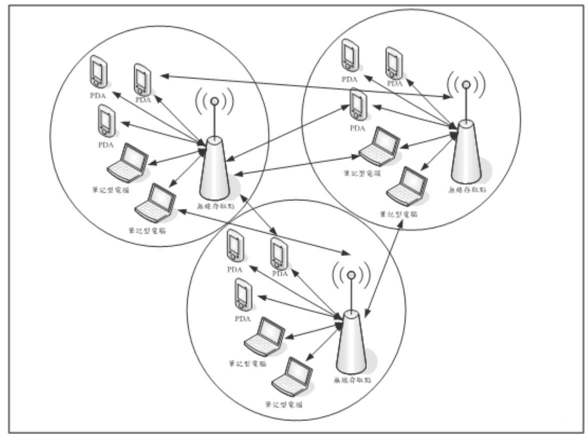


Fig. 2-3 ESS, Infrastructure

2.2 Positioning Technology

Through mobile positioning technology, the users can receive on-line information via mobile devices in the event of large-scale indoor activities. When a user enters into a certain area, his/her mobile device will display immediately the relevant information about this area. Such system acts as a personalized tour guide manual, which can provide the users with location-related information services, in addition to the user-specific location information in such environment.

The wireless communication technologies currently available for positioning include: GPS, GSM, IEEE802.11 Series WLAN (Wireless Local Area Network), Bluetooth and Infrared (Li-de Zhou, 2003), as listed in Tab. 2-1. Owing to limited range of indoor positioning, a higher degree of positioning accuracy shall be required to grasp the accurate location of users and offer correct information for reference. GPS, an existing system with a highest degree of positioning accuracy, is not taken into account for this system as GPS satellite signal cannot be received within an indoor environment. Also, owing to excessive tolerance of three-point positioning through GSM-based mobile access point, it is not suitable for indoor mobile positioning.

If Bluetooth or Infrared technology of excellent positioning accuracy is applied for indoor positioning, the requirement of mobile networking cannot be satisfied due to small signal coverage of these two technologies featuring only wireless networking.

IEEE802.11 Series WLAN, a leading technology of small-scale wireless networking, features cost-effectiveness and ease of operation. As most of access points are set up within indoor environment and present strong signal intensity of wide coverage, there is no problem of visual display of infrared and limited operating range. So, IEEE 802.11 Series WLAN is now a best solution for a balance between accuracy and mobility over indoor and outdoor positioning applications.

Tab. 2-1 Positioning technology（Kao, 2003; Chang, 2004）

Tech. / Item	Global Position System（GPS）	GSM Location Service	WLAN（IEEE 802.11Series）	Bluetooth	Infrared
Signal Range	Wide Area	Wide Area	Micro Area	Pico Area	Pico Area
Networking Connectivity	Good	Good	Good	Low	Low
Positioning Accuracy	N/A inside of building	Low	Medium	High	High
Signal Error Rate	N/A inside of building	High	Medium	Low	Low
Technology Complexity	High	High	Medium	Medium	Medium
Power consumption	Low	Low	Medium	Medium	Low
HW Cost	High	High	Medium	High	Low

2.3 Path Loss Characteristics of Radio Wave

IEEE802.11 Series WLAN is based on a transmission medium of radio wave. During transmission of radio wave, the intensity of radio wave will gradually decline with the increase of distance, as in the case of light intensity. The declined intensity of signal is called Path Loss. The remoter the mobile devices are located away from the access points, the weaker the received signal. According to the formula of physical theory, there is an inverse correlation between the intensity of radio wave and the square of distance/ frequency. As shown in Fig. 2-4, the signal loss will become bigger in the case of increasing distance or frequency. (You, 2003)

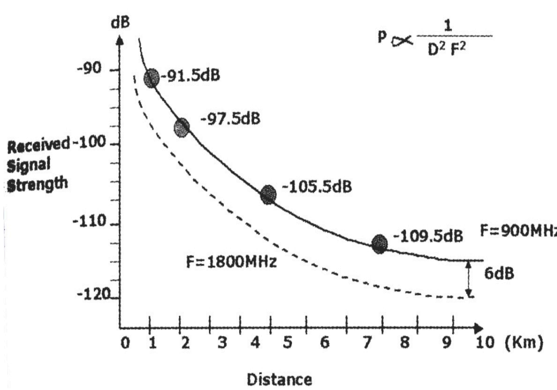


Fig. 2-4 Path Loss Characteristics of Radio Wave

According to the formula of radio wave transmission, the path loss between the emission point and receiving point of radio wave is: Path Loss = 32.4 + 20log F (MHz) + 20log D (Km), where F is the frequency of radio wave transmission, D is the distance between two points. To cite an access point with a radius of 10Km, the transmission frequency of antenna is 900MHz, and the transmitting power is 0dBm. According to the formula, the signal loss is reduced to 92dBm after 1 mile; the signal intensity will decline with the increase of distance as shown in Fig. 2.3. When the transmitting frequency of antenna varies from 900MHz to 1800MHz, the loss will be quadrupled if the frequency is doubled owing to the direct proportion between the loss and square of varying frequency. It can be found from this formula that the signal intensity will decline by 6dBm. Keeping in view of 2.4GHz applied by IEEE 802.11b, path loss is calculated to be approx. 74 dBm when the users have a distance of 50m with the access points. (You, 2003)

2.4 Methods of Wireless Positioning

2.4.1 Table-Based Method

Set up sampling points uniformly within the buildings for indoor positioning, then collect and record the signal intensity of every access point. So, every point is available with data of (x, y, ss_1, ss_2... SS_i), where x and y are locations coordinative of a sampling point, SS_i is the signal intensity received from every access point. Next, prepare a table of database. When intended for positioning, you may measure the signal intensity received from every access point, and compare it with the signal of every access point in the database. Then, calculate the minimum value via

$$\sqrt{(SS_1' - SS_1)^2 + (SS_2' - SS_2)^2 + \ldots + (SS_i' - SS_i)^2}$$

, (x, y) in the data is represented by the existing location of a user. Based on an experiment field of 8x8 sq. m., you (2003) set up sampling points from the crossing point of two lines in a space of 1m.

2.4.2 Multiple Nearest Neighbors Method

Continuing from Table-based positioning method, Table-based method is suited for finding an optimum point, while this method is used to add all coordinate values of nearest points for averaging, whereby the average value is an estimated location, there are three nearest neighboring points, such as N_1, N_2 and N_3, where T is a real location, G is an estimated one, so the average value is more close to the real location. (You, 2003)

2.4.3 Curve Fitting Method

According to the spherical signal distribution theory, it is required to measure the signal value when the users enter into the signal coverage of access points, where the users are located at the intersection of two access points, as shown in Fig. 2-5. (You, 2003)

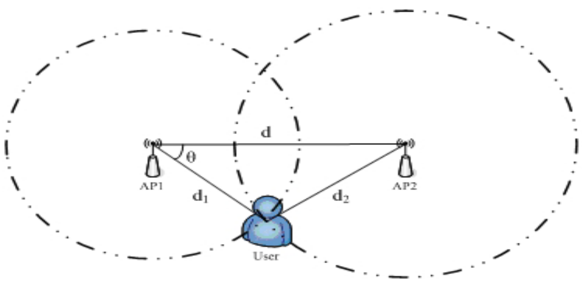


Fig. 2-5 Positions BTW User and APs

It can be learnt from Fig. 2-5 that, the users and two access points can constitute a triangle. If assuming a known d is the distance between two access points, d1 and d2 are distance values between them. If angle θ is calculated from formula θ, the location of users is made clear as the spatial location of mobile devices can be identified by angle θ. (You, 2003)

d1=0.0084 * ss2 -- 1.5299 * ss + 70.6028

d2=0.0086 * ss2 -- 1.5299 * ss + 66.7054

$$\theta = \cos^{-1}\left(\frac{d_2^2 - d_1^2 - d^2}{-2 \cdot d_1 \cdot d}\right)$$

3. Method of Experiment Design

3.1 Testing Environment

Within an area of 1200 sq. m., this system will set up an unobstructed testing environment for ESS-based network services, whereby the users can access to various services of this system. Additionally, when determining the configurations of access points for positioning purpose, a key principle must be taken into account, namely: the signal of over three access points must be received on any occasion at any location within the range of system service, as shown in Fig. 3-1.

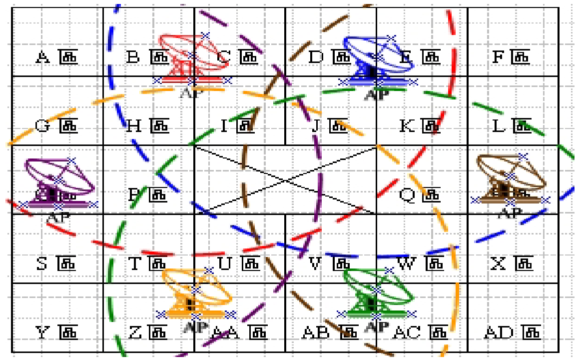


Fig. 3-1 Testing Environment

After setting up the range of system service in conformity to the specifications, divide the total area into some blocks of 40 sq. m., and measure the distance between every block and APs, then record the values/data and finally set up a “location database”.

3.2 System Architecture

This system adopts a Client-Server framework, which generally comprises three major system components: (1) Mobile Device User (MDU); (2) Location Information Server (LIS); (3) Service Provide Server (SPS). The functions of various components are described below:

(1) Mobile Device User: equipped with an on-board mobile device incorporating this system’s development procedure, liable for transmitting actively or passively the received signal of access points and user demands to the rear-end servers.

(2) Location Information Server: liable for positioning of the users’ location, comparing the received data of MDU to the data record of the database, and transmitting it to SIS database after calculating the users’ location.

(3) Service Provide Server: liable for responding to the demands of users. Through the data of users’ location received by LIS, it can transmit the location-related information to MDU according to the existing location of users, with the entire framework shown in Fig. 3-2.

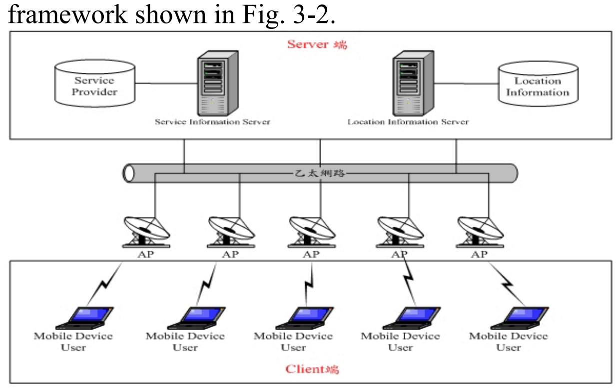


Fig. 3-2 System Architectures

3.3 Positioning Flow Design

If two APs are used for positioning, only left and right locations can be identified. Through the feature of radio wave's path loss, this three-point method can position correctly the direction of location, and then calculate the existing location of users. In this system, three APs are used for positioning of users' location. Based on extended service set (ESS) occurred between APs, the intersection set could definitely identify AP service location of users if three APs are used to position the service range. Therefore, positioning via three APs is more accurate than that via two APs, as shown in Fig. 3-3.

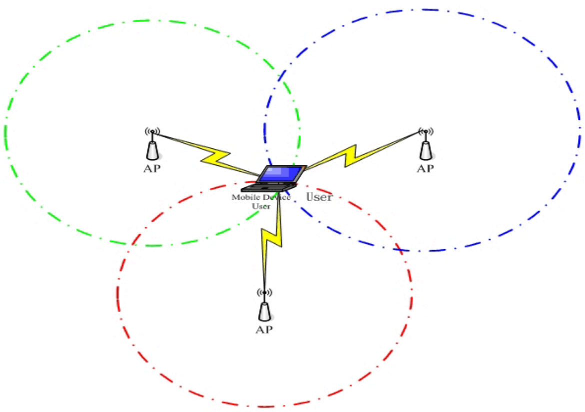


Fig. 3-3 Three Points Positioning

(1) When linked to wireless network, the users can measure the signal of electronic wave via NetStumbler v0.4.0 software, and transmit the received signal of Access Point to the rear-end servers for calculation.

(2) Through the relationship between signal intensity and distance, the servers can calculate the distance between the users and access points from the degree of signal loss.

(3) With the aid of three-point positioning method, collect three access points of strongest signal intensity from all received signals, and measure the intensity distance, then compare it with the location database to measure the coordinate values, thereby calculating the existing location of users, as shown in Figs. 3-4.

3.5 Location Base Tour Guide Service

3.5.1 Performance Analysis

This system can extend support to academic research, but also offer convenient, real-time and efficient application services in commercial application, as listed in Tab. 3-1. Compare traditional campus tour guide mode with location-based campus tour guide system, information provision and cost of tour guide will be better.

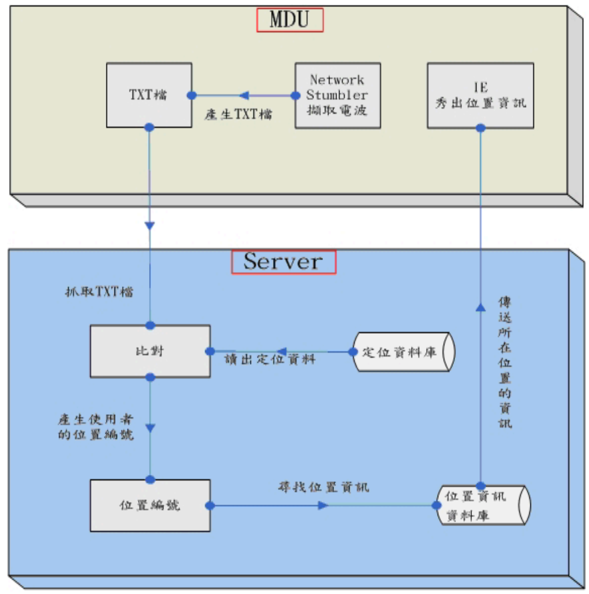


Fig. 3-4 Positioning Flows

4. Results and Conclusions

4.1 Location Base Tour Guide Service System

The WLAN-based campus positioning& tour guide system, which is operated by the visitors through mobile devices, is described below as shown from Fig. 4-1 to Fig. 4-2

4-2 Conclusions

Recently, a burgeoning move towards M from E has gradually taken shape all over the environment. Gartner expected that, the location sensing capability would become an integral function of application software and services by 2008. This will help create innovative processes and new business opportunities, leading to a sharp increase of customers of mobile positioning services. Amongst 1.5 billion customers of mobile services in 2005, 40% of them are customers of mobile positioning services. In the future, the environment will be remodeled to "Location-aware" from existing "Always on".

Tab. 3-1 Analysis of Positioning Service System

Service	Description of Application Service	System Performance
Broadcast	According to existing location of users, provide actively the users with latest information of nearest divisions or classrooms.	In comparison to traditional tour guide manual for freshmen, the universities can save the cost of additional print. It also offers user-specific information.
Personal Finding	Through wireless positioning know-how, record automatically the latest location information of divisions received by users, e.g. the parents may find	Also applied to person-searching service for indoor exhibition. For example, if a child is lost in the exhibition, the position person-searching

	out the location of children if they don't know the exact classroom of the children.	system can lend a support.
Service	Description of Application Service	System Performance
Tour Guide	Provide the route tour guide service covering entire campus. If a visitor wants to visit a building but doesn't know its route, the route tour guide service system can be used to guide the visitor for tour guide.	Applied to the positioning of stalls within a department store. The users with mobile devices can use the electronic tour guide map of this system to display the correct location of swimsuit block, etc, and get access to convenient, real-time and high-quality services.

The outdoor LBS service system has been launched into the market over a period of time, but indoor type hasn't yet emerged. This system has developed an integrated service information platform featuring "positioning function" and "suitable for indoor applications". After completion of system development, an on-line test was implemented against campus tour guide. Firstly, the campus tour guide information is integrated into the system service in a manner to enable it to serve as a campus tour guide system featuring positioning functions. In the presence of some visitors, no personnel are required to be designated by the Institute for guidance. Meanwhile, only mobile devices shall be provided to the visitors for a free movement in the campus. According to the location of visitors, this system can automatically transmit the location-related information, just like a personalized narrator. Once upon completion of successful test, diversified value-added services and applications shall be incorporated to promote the commercial value of this system. The achievements of this system are as follows:

(1) An indoor "wireless positioning system", which can control the latest location of users with aid of accurate positioning capability, has been successfully designed.

(2) An integrated "service position server" with expandability is provided for use with diversified value-added services and applications. Next, based on the location information of users obtained by "wireless positioning system", it can actively or passively transmit location-related services to the users.

(3) Take "campus tour guide" for example, a campus tour guide information system with positioning function is designed in order to materialize the positioning service functions of this system. .

References

1. Chang, C. Y., "A Hierarchical Indoor Positioning System and Location Based Services," National Centre University, Master thesis, 2004.
2. Comer, Douglas E., Internetworking with TCP/IP Principles, Protocols, and Architectures, Prentice Hall Inc., 2001.
3. Kao, J. L., "Design and Implementation of an IEEE 802.11-Based Location Service Platform," National Cheng Kung University, Master thesis, 2003.
4. Swaminatha, Tara M. and R. Elden Charles, Wireless Security and Privacy: Best Practices and Design Techniques, Addison Wesley Professional Inc., 2003.
5. You, S. W., "Research on using WLAN signals for indoor positioning," National Taiwan Technology University, Master thesis, 2003.

Interface of
Location Base Tour Guide Service System

Fig. 4-1 The System User Interface

Fig. 4-2 Positioning Results

Adaptive Multimedia Content Model in Learning Content Management System

Lawrence Y. Deng
Dept. of Computer Science and Information Engineering
St. John's Technology University
Tamsui, Taipei, R.O.C.
Lawrence@mail.sjsmit.edu.tw

Abstract **Multimedia presentation systems require flexible support for the modeling of multimedia content models. Many presentation systems provide the synchronized, sequential or concurrent, and possible interactive, transfer of streams multimedia data such as audio, video, text and annotations using with network facilities. However, our through investigation of exist standards and applications for multimedia documents models such as HTML, MHEG, SMIL, HyTime, RealPlay and MS Windows Media let us to find that these standards and applications models do not provide adequate support for reuse and adaptation. Consequently, we propose a new approach for the modeling of reusable and adaptable multimedia content. We developed a comprehensive system for advanced multimedia content production: support for recording the presentation, retrieving the content, summarizing the presentation, weaving the presentation and customizing the representation. This approach significantly impacts and supports the multimedia presentation authoring processes in terms of methodology and commercial aspects.**

Index Terms*—*Multimedia Presentation, Media Synchronization, Adaptive Presentation, Multimedia Content Model, Multi-level Content Tree

I. INTRODUCTION

Multimedia content model among the network are most often used in many communication services. Examples of those applications include video-on demand, interactive TV and the communication tools on a distance learning system and so on. To control and demonstrate different types of multimedia objects is one of important functions in distributed multimedia presentation system. There are some important design considerations during to construct multimedia presentation. First is how to model and describe the media properties during presentation period. A multimedia presentation basic requirements including to demonstrate the media's spatial, temporal, and user interaction properties. Second is the presentation advance requirement. User always wants alternative adaptation operation facilities, such as, authoring, retrieving, abstracting/summarizing, or even rearranging the presentation performances. Unfortunately, we were rarely to see the existing presentation software could satisfy these advance requirements.

We have been working on multimedia presentation for several years in related popular commercial or nonprofit projects [1, 2, 3, 4, 5]. Over the past year, we have been compared and identified methods to work more closely together and to provide with an adaptability and reusability. An effective presentation design procedure should not only involve sequential flow of actions, but also parallel/concurrent and user interactive actions. Additionally, the design includes a number of high-level concerns, such as goals and focus of the presentation, the user's context and current task, and the media selection to represent the information in a way that corresponds to these concerns.

The first step of the research is to design a multimedia content model, which is built upon with the existing models. A multimedia content model is a model comprised of information coded in at least one time-dependent medium (e.g. video, audio…) and in one time-independent medium (e.g. text, image…). The corresponding multimedia document architecture and their relations are described in [6]. Multimedia document architecture demonstrates the relationships among the individual components represented as models. It includes the presentation model, manipulation model and representation model. The presentation model illustrates the media elements how they to be processed during running time. The manipulation model describes all the possible operations allowed for creation, change and deletion of multimedia information. The representation model not only defines the protocols for exchanging this information among different computers but also the formats for storing the data. It contains the relations between the individual media elements which need to be considered during presentation. Structure implicates the basic requirements and advanced requirements while these models operate their functions.

This remain paragraph is organized as follows. Section II discussed the multimedia presentation design consideration and background knowledge with the main properties of the user-concerned adaptation presentation model. Section III gives the formal framework and the related adaptation operation for a detailed understanding of the multimedia content model. Section IV summarizes our work and gives an outlook to ongoing and future work.

II. Background Knowledge on Content Model Requirements

As foregoing discussion, multimedia presentation systems need to integrate the presentation module, representation module and manipulation module. There some basic and advanced presentation requirements need to constitute and comprehend. The basic presentation contains the spatial, temporal and user interaction within synchronization phase. The work [7] presented a scheme for intra- and inter- stream synchronization of orchestra multimedia. Spatial-temporal relations of a multimedia presentation were discussed in [8]. The DEFSM model [9, 10] not only discussed the intra-medium synchronization and inter-medium synchronization but also provided the VCR-like (i.e. reverse, skip, freeze-restart and scale) user interactions that allow user to modify the presentation configuration at any time during the presentation. The advanced presentation includes the usabilities and user-concerned adaptabilities. Following paragraph would like to demonstrate the basic presentation requirements: temporal, spatial and user interactions models and discuss the advanced presentation requirements: media usability and user-concerned adaptabilities.

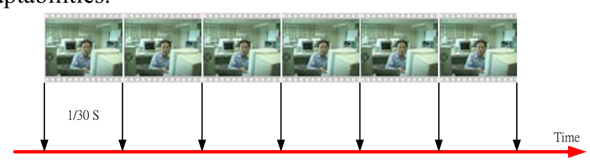


Figure 2: An example of the intra-object synchronization.

Temporal relations define the time dependencies between media elements. The intra-medium synchronization refers to the time relation different presentation elements of one time-dependent media element. Figure 2 shows an example of the time relation between the single video frames of a certain sequence. The inter-medium synchronization refers to the synchronization between media elements. Figure 3 illustrates an example of the time relations of a multimedia synchronization. This example is a lecturer presentation with video, slides (images) and possible annotation tools supported.

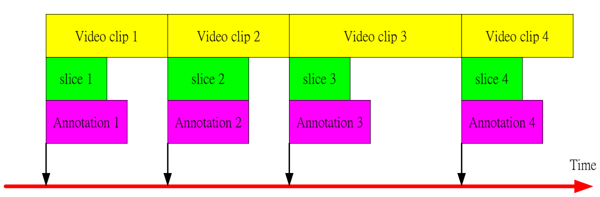


Figure 3: An example of the inter-object synchronization.

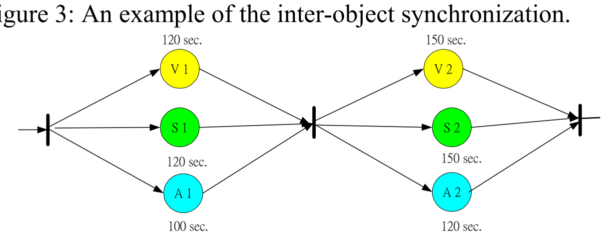


Figure 4: A Petri net representation of the example figure 3.

Figure 4 is the case of event-based synchronization with Petri net model representation, which represents the example of example 3. All the presentation operations about the user made are initiated by synchronization events (such as the mouse click events, windows messages for annotation/comments tools…).

Additional extensions have been proposed, and this has led to the following types of Petri nets: the timed Petri net, the stochastic Petri net, colored Petri net, and object-related Petri net [11, 12]. The "Object Composition Petri Net"(OCPN) and the "extended Object Composition Petri Net"(XOCPN) are two graphic-based models that propose synchronous theoretical for multimedia. The OCPN is a comprehensive model for specifying timing relations among multimedia data. The XOCPN can specify temporal relationships for the presentation of pre-orchestrated multimedia data, and to set up channels according to the required Qos of the data. These two models lack methods to describe the details of synchronization across distributed platforms and do not deal with the presentation schedule change caused by user interactions in interactive multimedia systems.

In this article, we use the extended timed Petri net to construct the web operations on a multimedia presentation system. When multimedia presentation objects are represented on the system, we have to consider different situations of multimedia objects such as asynchronous operations, time scheduling, and flow-control. In addition to system operations, dynamical and unpredictable operations of users are important issues. Thus, we can apply characteristic of Petri net to implement our mechanism and study the theory.

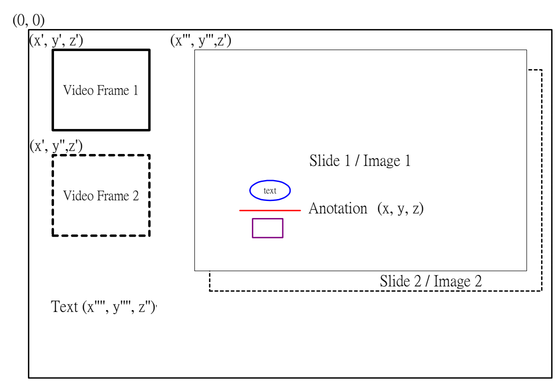


Figure 5: A spatial relations layout the activation media elements in a window.

The spatial relations can layout the presentation media elements and their relationship. There are two possibilities in spatial relations: absolute positioning and relative positioning. The upper right corner (original referent point) of the window is defined as position (0, 0). Both absolute and relative positioning consults this original point to specify their X-axis, Y-axis and X-axis index respectively. Figure 5 illustrated a spatial relations layout the activation media elements in a window.

With user interactions, he or she can (a) reverse the flow of the presentation; (b) skip forward or backward to a specified presentation segment; (c) freeze and then restart the presentation, and (d) scale the speed up or down. It provides such user interactions are that they are issued dynamically and unpredictably during the presentation.

Reuse capability is the necessary in a multimedia presentation system. There are some different levels of reusabilities: reuse of complete multimedia presentation document, reuse of segments of multimedia as single scenes or a lectured chapter/unit, and reuse of individual atomic media elements such as a video, audio, slides or text and parts of those media elements such as a scene of a video clip.

Especially, multimedia presentation system should provide adaptability during authoring or viewing the presentation. For example, user can adapt the bandwidth/qualities of the performance for different devices/services needs (e.g. PDA, notebook, PC…) during the authoring process. During the presentation, the multimedia presentation system should offer retrieving, abstracting, additional adapting (e.g. select channels, time specified), weaving, and customizing operation. These adaptive operations in a multimedia presentation system always satisfy for user context.

III. Formal Representation of Our Approached

We define multimedia objects representation based on the characteristics of the Petri net. As a graphical tool of Petri net, the followings are basic properties of a Petri net and the description of multimedia objects:

Definition 3.1: A presentation Petri net is a 8-tuple,

PN = (P, T, A, K, D, C, F, ID) where:

- P = {P1, P2, ………, Pm} is a finite set of places,
- T = {T1, T2, ……….., Ti} is a finite set and a sequence of transitions,
- $A \subseteq (P \times T) \cup (T \times P)$ is a set of arcs,
- K = {α, β, ……..ζ} $\in$ String is a set of Keyword,
- D :P$\rightarrow$ {0, 1, 2, ……} is the degree of tags,
- C = {A, B, Z}is a set of channel,
- F : P$\rightarrow$ {0, 1, 2, ……} is the frequency of media objects to be accessed,
- ID: P$\rightarrow$ {0, 1, 2, ……} is the identifier of a media element,
- $P \cap T = \varnothing$ and $P \cup T \neq \varnothing$.

The generic components of Petri net include a finite set of places and a finite set of transitions. Petri net is a finite bipartite graph. Its places are linked with transitions in turn are connected to the output places. For a given place, there are input and output transitions defined.

Definition 3.2: The basic unit of media content is the atomic media element. An atomic media element is an instance of media type. The available media types in our model contain video, audio, image, text, layout, and the windows messages (annotation). The atomic media element is also represented the media channel of what they used in run time environment respectively.

Definition 3.3: A place P contains an atomic media element, a complex media element, a specific operator element to construct the temporal, spatial relationships, or serve for the retrieval and the user's specification adaptation. We write P= (K, D, C, F, ID] to denote such a place.

Definition 3.4: A transaction T involves the temporal operator elements. They are three types of temporal operator elements: Ts(seq), Tp(par), and Te(excl). Ts = {T1, T2, …, Ti} is a finite set and a sequence of transitions, the temporal operator element Ts bound the presentation media element in sequence. Following figure illustrates the Ts operator element starts the sequential presentation PN with P1 and ends with the presentation media element Pn. Tp = {T1, T2, ……….., Tj} is a finite set and a parallel of transitions, the temporal operator element Tp render the presentation media elements in parallel. In this temporal model, this didn't means that they get effect at the same time but they share a global timebase, any or all of son can be active at specific time that the parent par is active. Te = {T_1, T_2, ……….., T_k} is a finite set and a *exclusive* of transitions. Only one of the sons of the excl tags can be active at a time. The period of the excl presentation depends on the begin attributed of the excl's son. Each place will have an event-based start condition(e.g. begin="IDname.activeEvent") that lets one of the children start on user's demand.

Definition 3.5: A place P (atomic media element) involves a spatial operator element. A spatial operator element applies the presentation place P with specific absolute/relative position parameter: x-axis index, y-axis index, z-index, height, and width. The x-axis index is described as left:100px or left:50%. The y-axis index is described as top:100px or top:50%. The z-axis index is a vertical parameter; it can control and stack the presentation media. The z-axis index is described as z-index = n, $n \in N$. (the number n bigger the specific presentation element P topper).

For the most part, a multimedia presentation must be produced a great quantity of data capacity. In most existing presentation system, user didn't to be endowed with retrieving, abstracting, additional adapting (e.g. select channels, time specified), weaving or customizing facilities during the presentation. These above-mentioned presentation system facilities are user-concerned in the available system resources. In the section, we assume that multimedia content model must offer the possibility to represent alternatives and categorically to conform to the dynamic user-context.

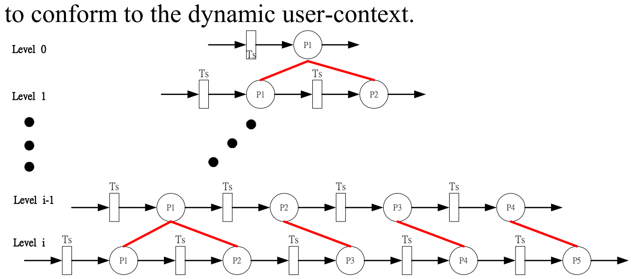


Figure 6: An example of a presentation to represent to be a content-tree, where Pi is denoted as a presentation segment.

Due to the unpredictable changes in the dynamic user context, the multimedia content model must be able to model alternatives not only on the level of atomic media elements but also on the level of user specific segment. Given a multimedia document with foregoing definitions, the corresponding multi-level content tree can be constructed, as shown in Figure 6. A multimedia presentation (e.g. collection of text, video, audio, image…etc.) can be taken as some kinds of sequence fashion. The multiple level content tree approach may be used to arrive at an efficient summarizing method.

A content tree is a finite set of one or more nodes such that there is a particularly designated node called the root. The level of a node is defined by initially letting the root be at level 0. If a node is at level q, then its children are at level q+1. Since a node is composed of a presentation segment, the siblings with the order from left to right represent a presentation with some sequence fashion. The higher level gives the longer presentation.

By retrieval we mean the multimedia presentation system can satisfy the storage and retrieval requirement s of a very large number of atomic media object (possibly hundreds of millions) where a presentation can have a storage requirement of several hundred megabytes. Therefore, this is closely impossible to query in multimedia presentation system by using content-based image/video retrieval techniques. In our approach, we defined the attributes "Keyword" to achieve user demand. Keyword attributed can be extracted form the title or presenter's specified of the presentation slides/transparencies. Queries are expressed in terms of high-level declarative constructs that allow users to qualify what they want to retrieve from the persistent multimedia presentation system. The retrieval definition is defined as follow.

Definition 3.6: The retrieving operation, ρk(PN'{P'1, P'2, …, P'm}, PN{P1, P2, …, Pn }) extracts from PN{P1, P2, …, Pn } all the keyword k of the presentation media place P'i that are similar to PN'{P1, P2, …, Pm} with respect to the similarities threshold keywords.

Let the set of keyword k' 1 ∈ P' 1, k' 2 ∈ P' 2,….. ,k'm ∈ P'm, where P i ∈PN', and k 1 ∈ P 1, k 2 ∈ P 2,….. ,kn ∈ Pn, where P' i ∈PN.

ρk(PN'{k'1, k'2, …, k'm}, PN{k1, k2, …, kn }) = PN{k'1, k'2, …, P'm}

→ ρk(PN'{P'1, P'2, …, P'm}, PN{P1, P2, …, Pn }) = PN{P'1, P'2, …, P'm}

In abstraction operation, we defined the attributes "degree" to achieve user demand. Degree attributed can be represented by the subtitle or presenter's specified of the presentation slides/transparencies. Abstractions are expressed in terms of high-level declarative constructs that allow users to qualify what they want to abstract from the persistent multimedia presentation system. The abstraction definition is defined as follow.

Definition 3.7: The abstracting operation, αd (PN{P1, P2, …, Pn }) extracts all the presentation media place Pi with degree d.

Let the set of degree d 1 ∈ P 1, k 2 ∈ P 2,….. ,kn ∈ Pn, where P i ∈PN.

αd (PN{d1, d2, …, n }) = PN{d'1, '2, …, P'm}

→ αd (PN{P1, P2, …, Pn }) = PN{P'1, P'2, …, P'm}

where the degree of P'i in PN{P'1, P'2, …, P'm} is equal to d.

In additional adaptation operation, there are two additional adaptation operations in our presentation model: channel selection and time specification adaptation. Firstly, we defined the attributes "channel" to achieve user demand. The purpose of a channel is to be a media grouping abstraction for a set of media items that display some common media attributes. The channel provides a logical thread upon which media objects can be placed. This thread can be switch on or off during the presentation based on the needs of user or user adaptation agent. The channel abstraction can be to satisfy user-concerned adaptation, but it can also to be used by the presentation on-line system to select fractional Qos adaptation of multimedia content alternatives. The channel abstraction definition is defined as follow.

Definition 3.8.a: The channel abstracting operation, γc (PN{P1, P2, …, Pn }) extracts all the presentation media place Pi with degree c.

Let the set of degree c 1 ∈ P 1, k 2 ∈ P 2,….. ,kn ∈ Pn, where P i ∈PN.

γc (PN{d1, d2, …, n }) = PN{d'1, '2, …, P'm}

→ γc (PN{P1, P2, …, Pn }) = PN{P'1, P'2, …, P'm}

,where the channel of P'i in PN{P'1, P'2, …, P'm} is equal to c.

Another, we use the media element attributes "duration" to achieve user demand (i.e. time specification) while she or he can designate the presentation period. As we known, the presentation can be represented to be a content-tree in our foregoing definition. In a content-tree, the multimedia presentation is a multi-level of sequence fashions. The higher level gives the longer presentation. After the user specified the "time" of what they desire, our presentation generator will compare it with the presentation period in the different level sequence fashions. Then, the appropriate level of sequence fashions will be selected. The remained processes are same as the abstraction operation.

Definition 3.8.b: The time abstraction operation.

Let the time of user specified is defined as τ; the period of sequence fashion is defined as ϕ; the set of degree d 1 ∈ P 1, k 2 ∈ P 2,….. ,kn ∈ Pn, where P i ∈PN.

```
Process:
FOR i=1 to i<=d DO
IF (ϕi<τ AND τ<ϕi+1) THEN
   d=i
Return d
   END IF
  End FOR
End Process
```

And, αd (PN{d1, d2, …, n }) = PN{d'1, '2, …, P'm}

→ αd (PN{P1, P2, …, Pn }) = PN{P'1, P'2, …, P'm}

where the degree of P'i in PN{P'1, P'2, …, P'm} is equal to d.

Definition 3.9: The weave operation, ωselect (PN{P1, P2, …, Pn }) weaves all the presentation media place Pi into a new PN' {P'1, P'2, …, P'm}. The designated select is evaluated from the user adaptive profiles.

Let the set of dynamic attribute select $s_1 \in P_1$, $s_2 \in P_2$,… ,$s_n \in P_n$, where $P_i \in$PN, $s_i \in$ user adaptive profiles.

Process:

FOR i=1 TO i<=n DO

IF (si= = TRUE) THEN

ωselect (PN{P1, P2, …, Pn }) = PN'{P'1, P'2, …, P'm},

END IF

End FOR

End Process

where the P'1, P'2, …, P'm $\in$ PN.

As above mentioned, the weave, additional adaptation, abstract, and retrieval operation can provide an alternative adaptation for the use-concerned specification. After user decided to perform these user-specified operations, the customization operation provides user advanced to store the new presentation vision to be a persistent presentation media. The server presentation generator will modify the multimedia content document with certain constrains (satisfied the user' adaptive profile and the multimedia content model). The customization operation definition is defined as follow.

Definition 3.10: The customization operation, θselect (PN{P1, P2, …, Pn }) weaves all the presentation media place Pi into a new PN' {P'1, P'2, …, P'm}. The designated select is evaluated from the user adaptive profiles.

Let the set of dynamic attribute select $s_1 \in P_1$, $s_2 \in P_2$,. ,$s_n \in P_n$, where $P_i \in$PN; $s_i \in$ user adaptive profiles; PN1, PN2 ….PN k $\subseteq$ multimedia presentation system database.

FOR i=1 to i<=n DO

IF (si= = TRUE) THEN

θselect (PN1{P1, P2, …, Pa }, PN2{P1, P2, …, Pb },….. PNk{P1, P2, …, Pn })

= PN'{P'1, P'2, …, P'm},

where the P'1, P'2, …, P'm $\in$ PNi.

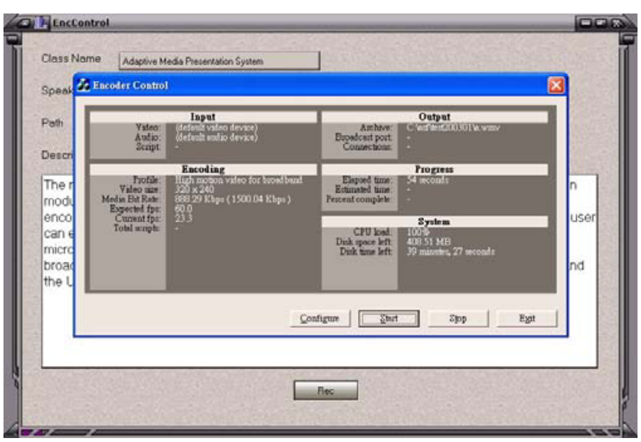


Figure 7: encoder control window: this windows shows the input devices configuration and output devices configuration steps for the recording operation

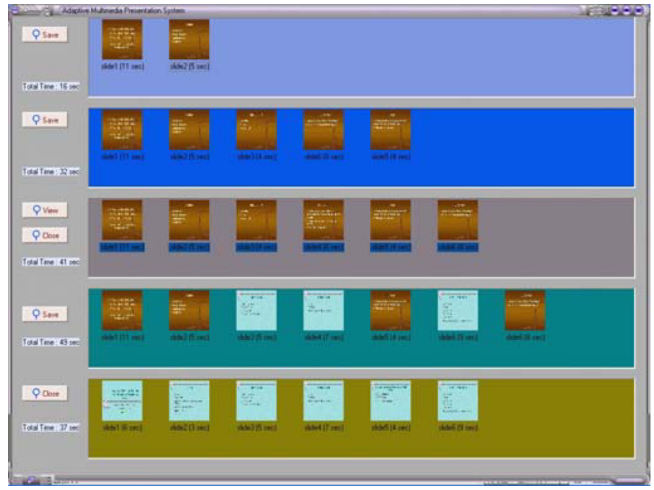


Figure 8: adaptive media presentation windows, user can drag and drop the media object for weaving, addition adaptation, abstracting, retrieval operations as user-concerned specification.

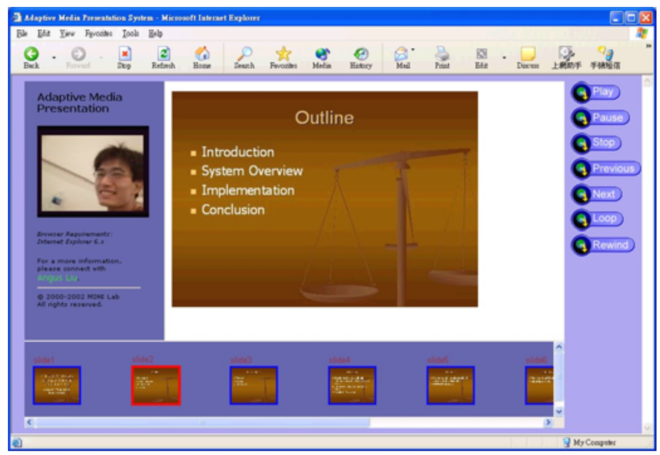


Figure 9: web-based presentation windows

IV. Conclusion

In this article, we not only proposed a well-defined multimedia content model but also offered a framework of web-based multimedia presentation system. In order to model the fundamental presentation requirements clearly, we begin to use the extended timed Petri Net as the media synchronization model, followed by the formalized definitions of user-concerned adaptive adaptation operation. Thus, the configuration and the operation steps of the multimedia presentation system are clear and definite. The final goal of our approach is to provide a feasible multimedia content model and the unequivocal framework to developer as guiding principle or policy. We hope that this approach can be used to the general purposed multimedia presentation system such as distance learning, enterprise training, commercial advertisement, and others.

References

[1] Multimedia Micro-University Virtual Society, http://www.mine.tku.edu.tw/mmvs

[2] Timothy K. Shih, Lawrence Y. Deng, J. H. Wang, and Andy Ay-Hwa Liou, "Maintaining Persistent Look-and-feel for Roaming Student with Mobile Agent in Distance Learning," in Proceeding of the 7th International Conference on Distributed Multimedia Systems (MMS' 2001), Taipei, Taiwan, September 2001.

[3] Timothy K. Shih, and Lawrence Y. Deng, "Distributed Multimedia Presentation with Floor Control Mechanisms in a Distance Learning System," in Proceeding of the 2000 International Conference on Information Society in the 21 Century: Emerging Technologies and New Challenges, Japan, November 2000.

[4] Flor Chia-I Chang, Te-Hua Wang, Yu-Wei Chan, Lawrence Y. Deng, I. Lee and Y. Wang, "Internet-based Distance Education: an Agent-based Distance Learning Project in Tamkang University as an Example," in Proceeding of the 7th International Conference on Distributed Multimedia Systems (MMS' 2001), Taipei, Taiwan, September 2001.

[5] Chang S. K.; Hassanein E.; Hsieh C. Y., "A Multimedia Micro-University", IEEE Multimedia Magazine, Vol. 5, No. 3, July-September 1998, pp.60-68.

[6] Ralf Steinmetz and Klara Nahrstedt, "Multimedia: Computing, Communications and Applicatiions", Prentice Hall Inc., pp. 481-542, 1995.

[7] A. Bierasack and W. Geyer, "Synchronized Delivery and Layout of Distributed Stored Multimedia Streams", Journal of Multimedia System, vol. 7, No. 1, pp. 70-90, 1999.

[8] C. M. Huang, J. S. Chen, C. H. Lin and C. Wang, "MING-I: A Distributed Interactive Multimedia document Development Mechanism", Journal of Multimedia System, Vol. 6, No. 6, pp. 316-333, 1998.

[9] C. M. Huang, C. Wang and C. H. Lin, "Interactive Multimedia Synchronization in the distributed environment Using the Formal Approach", IEE Proceedings of Software, Vol. 147, No. 4, pp.131-146, August 2000.

[10] C. M. Huang and C. Wang, "synchronization for Interactive Multimedia Presentation", IEEE Multimedia, Vol. 5, No. 4, pp. 44-62, 1998.

[11] J. L. Peterson, "Petri Net Theory and the Modeling of Systems," Englewood Cliffs, NJ: Prentice-Hall, Inc. 1981.

[12] [13] Richard Zurawsi and MengChu Zhou, "Petri Nets and Industrial Application: A Tutorial", IEEE Transactions on Industrisl Electronics, Vol. 41, No. 6, Dec. 1994.

An intelligent fuzzy filter design via Takagi-Sugeno Models

Fang-Ming Yu[1], Cheng-Ning Huang[2], and Hung-Yuan Chung[2]

[1]*St. John's & St. Mary's Institute of Technology, Department of computer science and information Engineering*

[2]*National Central University Department of Electrical Engineering*

Abstract

In this paper, we present a design procedure of intelligent fuzzy filter using Takagi-Sugeno fuzzy models. The proposed fuzzy filter has the intuitive design of Mamdani fuzzy system and the efficiency of T-S fuzzy model. This architecture is very suitable to design fuzzy controller by hardware realization and software programming. The simulation result proves the approximation between the original Mamdani fuzzy system and the proposed T-S fuzzy filter. In the future, we will apply the T-S fuzzy filter in practical application.

Keyword: Takagi-Sugeno fuzzy model, Mamdani fuzzy system, fuzzy filter

1. Introduction

Since Professor Lotfi Zadeh published the first ideas on fuzzy sets in the early 1960's, fuzzy logic (FL) has raised considerable attention. Nowadays, we can find that fuzzy logic has been applied successfully in different fields, such as robot manipulator[1], vehicle automation[2] and signal processing application[3]. Some researches utilize fuzzy logic to solve system identification problems[4]. The reason why people ardently devote themselves to fuzzy logic is very simple. Because fuzzy inference is based on the human knowledge and common sense, it is intuitive and easy to understand. The exact model of plant is not needed for fuzzy controller design. Furthermore, the fuzzy logic can be easily combined with classic control techniques and can be operated in coordination.

In spite of the advantage and popularity of fuzzy logic, it still has some considerable drawbacks. When we apply fuzzy logic in the interesting topics, the fuzzy reasoning algorithm cannot be easily implement on general-purpose microprocessor. This is because to describe the shape of membership function and to infer the linguistic if-then rules is a great challenge. Although we can achieve the fuzzy system by software implementation on the personal computer, it has long response time. The long computational time could raise real-time issue in control application. Costa et al identifies four implementations of fuzzy logic among the different hardware alternatives[5]. There are different hardware/software tradeoffs that must be considered in the design of fuzzy controllers. However, a universal solution for the implementation of fuzzy controller is still unavailable, at present.

In the 1980's, Japanese researcher Sugeno proposed the famous Takagi-Sugeno fuzzy model (T-S fuzzy model)[6]. The main feature of T-S fuzzy model is to express the local dynamics of each fuzzy implication rule by a polynomial. In this model, fuzzy rule consequent are assumed to be linear combination of the input variables, and the output is a convex combination of consequents, with coefficients that are the grades of membership function of the inputs in the antecedents. Comparing with the Mamdani fuzzy model, the T-S output membership function has been replaced by the linear (nonlinear) polynomial. This architecture provides the feasibility of stability analysis. It can also reduce the computational efforts of fuzzy logic. This is the main reason why Takagi-Sugeno's controllers are preferred for hardware implementation.

In this paper, we extend the T-S fuzzy model to a new form, called T-S fuzzy filter (TS-FF). The proposed fuzzy filter has a compact architecture, which can be approximated to the Mamdani or Sugeno fuzzy controller.

2. The proposed fuzzy inference model

In this section we give a brief introduction on the proposed fuzzy inference system, which is a modified from Takagi-Sugeno Fuzzy model, which can provide great efficiency in many applications. The original form of Takagi-Sugeno fuzzy model can be represented by the following general form [7]:

$$\text{IF } z_1(t) \text{ is } M_{i1} \text{ and ... and } z_p(t) \text{ is } M_{ip}$$
$$\text{THEN} \begin{cases} \mathrm{x}(\mathrm{t}+1) = \mathrm{A}_i x(t) + B_i u(t), \\ y(t) = C_i x(t) \end{cases} \quad \mathrm{i} = 1,2,\ldots,\mathrm{r}. \qquad (1)$$

Here, M is the fuzzy set and r is the number of the model rule. $Z(t) = [z_1(t) \quad z_2(t) \quad \ldots \quad z_p(t)]$ is the input vector and p is the number of input variables. Each fuzzy rule contains the linear consequent equation A_i, B_i and C_i, The linguistic label M_{ij} is the grade of membership function, which is described by $x_j(t)$ in the i-th μ_i. Given the input vector $Z(t)$ and $u(t)$, the result of the fuzzy inference are inferred as follows:

* Corresponding author: fmyeu@ntist.edu.tw

$$x(t+1)=\frac{\sum_{i=1}^{r} w_i(z(t))\cdot\{A_i x(t)+B_i u(t)\}}{\sum_{i=1}^{r} w_i(x(t))} \qquad (2)$$

$$=\sum_{i=1}^{r} h_i(z(t))\cdot\{A_i x(t)+B_i u(t)\}$$

$$y(t)=\frac{\sum_{i=1}^{r} w_i(z(t))\cdot C_i x(t)}{\sum_{i=1}^{r} w_i(x(t))}=\sum_{i=1}^{r} h_i(z(t))\cdot C_i x(t) \qquad (3)$$

where

$$w_i(z(t))=\prod_{j=1}^{p} M_{ij}(z_j(t)) \qquad (4)$$

$$h_i(z(t))=\frac{w_i(z(t))}{\sum_{i=1}^{r} w_i(z(t))} \qquad (5)$$

If we adopt T-S fuzzy structure to achieve a controller or to deal with signal, it is difficult to determine the system parameter A_i, B_i and C_i by experience. We have to construct the plant model previously and to estimate the system matrix of controller by parallel-distributed compensation (PDC) or other analytical methods. So we hope to keep the model-free advantage of Mamdani fuzzy system and to implement the controller by efficient T-S fuzzy model. First, we developed a Mamdani fuzzy system by human knowledge base. Then we construct a T-S fuzzy system to model the original developed Mamdani fuzzy system. In order to approximate the Mamdani fuzzy system, the T-S fuzzy structure is modified as follows:

IF $x_1(t)$ is M_{i1} and ... and $x_p(t)$ is M_{ip}
THEN $y(t)=C_i x(t)$ i = 1,2,..., r. (6)

$$y(t)=\frac{\sum_{i=1}^{r} w_i(x(t))\cdot C_i x(t)}{\sum_{i=1}^{r} w_i(x(t))}=\sum_{i=1}^{r} h_i(x(t))\cdot C_i x(t) \qquad (7)$$

$$w_i(x(t))=\prod_{j=1}^{p} M_{ij}(x_j(t)) \qquad (8)$$

$$h_i(x(t))=\frac{w_i(x(t))}{\sum_{i=1}^{r} w_i(x(t))} \qquad (9)$$

If we take the y and x as the fuzzy filter input and output separately. The weighted average of the each fuzzy rule $h_i(x(t))$ can be regard as fuzzy filter weighting. In the next section, we would indicate how to estimate the filter weighting.

3. The scheme of proposed fuzzy filter

In the pervious section, we give a brief introduction in T-S fuzzy model structure. In order to apply fuzzy filter in hardware implementation, we have to make some practical assumptions. First, we only consider the single input variable condition. If j=1, the equation (8) can be simplified as

$$w_i(x(t))=M_i(x(t)) \qquad (10)$$

Each linguistic label M_i is associated with a membership function, μ_i, which can be described by the L-R type fuzzy number expression:

$$\mu_i=\begin{cases}\frac{a_i-x}{\alpha_i}, a_i<x<m_i\\ 1, \quad m_i\le x\le n_i\\ \frac{x-b_i}{\beta_i}, n_i<x<\beta_i\end{cases} \qquad (11)$$

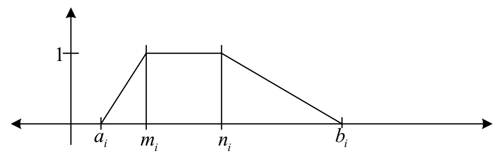


Figure 1 L-R type fuzzy number

Figure 1 illustrates the shape of L-R type fuzzy number expression, where a_i and b_i is the upper limit and lower limit of input membership function, respectively. α_i and β_i are the inverse of i-th trapezoid slope which are denoted as:

$$\alpha_i=\frac{1}{a_i-m_i} \quad \text{and} \quad \beta_i=\frac{1}{n_i-b_i} \qquad (12)$$

Furthermore, we simplify the membership as triangular shapes. In practical application the triangular membership function is easily and frequently to be used. So we can set $m_i=n_i$, the fuzzy number expression becomes

$$\mu_i=\begin{cases}\frac{a_i-x}{\alpha_i}, a_i<x<m_i\\ 1, \quad \mathrm{x=m_i}\\ \frac{x-b_i}{\beta_i}, m_i<x<\beta_i\end{cases} \qquad (13)$$

and value of α_i and β_i are simplified as

$$\alpha_i=\frac{1}{a_i-m_i} \quad \text{and} \quad \beta_i=\frac{1}{m_i-b_i}$$

The simplify input membership function of fuzzy filter is given in Figure 2.

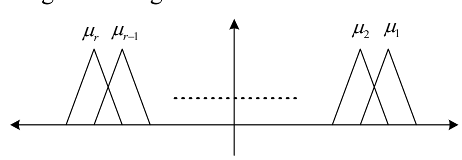


Figure 2 Input membership function of fuzzy filter

In order to meet the property of rule-base completeness and consistency, we let the triangular fuzzy number shape as $a_i=m_{i-1}$ and $b_i=m_{i+1}$. When the crisp input is fed into fuzzy filter, there are two membership functions at most could be fired at same time. Moreover, the defuzzification can be

simplified as $\sum_{i=1}^{r} w_i(x(t))=1$ (14)

Then $h_i(x(t)) = \frac{w_i(x(t))}{\sum_{i=1}^{r} w_i(x(t))} = w_i(x(t))$ (15)

Using the relation of eq.(10), the defuzzification of fuzzy filter is obtained as $h_i(x(t)) = M_i(x)$ (16)
Fuzzy filter result is performed by applying eq. (16) to eq. (7), which yields

$$y(t) = \sum_{i=1}^{r} M_i(x(t))C_i x(t) \quad (17)$$

From the above derivation, we found that the Mamdani output membership is characterized by $C_i x(t)$. C_i can be seen as a set of filter weighting and $M_i(z(t))$ is similar to the weighting adaptation algorithm.

3.1 Fuzzy filter weighting evaluating

In the following we shall discuss the filter weighting and adaptation algorithm. Owing to Mamdani fuzzy system is easy to design and has been widely applied in industrial application, we will begin with making a description of transforming a Mamdani fuzzy system to a sugeno fuzzy system.

For approximating Mamdani model, we adopt the same input membership function on sugeno and Mamdani model. The most different between Mamdani and Sugeno fuzzy model is the consequent of fuzzy rules. Now, the question is how to choose the parameter C_i so that sugeno model is similar to Mamdani model. For example, if the Mamdani adopt gaussian shape as output membership function, then the consequent of sugeno model can be expressed as:

$$f_i^j(x(t)) = \exp\left(\frac{-(x_i(t) - m_{ij})^2}{\sigma_{ij}^2}\right) \quad (18)$$

Nevertheless, the triangular or crisp output membership is more suitable for hardware implementation. The triangular shape fuzzy set is a non-continuous function, it is not easy to describe in the fuzzy consequent. However, we can make the defuzzification result of Sugeno and Mamdani model equal. Define the defuzzification value of sugeno model as C_i.

We can obtain $C_i = Defuzzify(\mu_i^{mad\_out})$ (19)

In this paper, we use the Center of Gravity defuzzification method to evaluate crisp defuzzication value. The COG method assumes that the crisp representative y^* is taken to be the geometrical center of the output fuzzy number. The defuzzified output is defined as

$$y^* = \frac{\sum_{i=1}^{r} y_i \mu_i^{mad\_out}(y_i)}{\sum_{i=1}^{r} \mu_i^{mad\_out}(y_i)} \quad (20)$$

3.2 Weighting adaptation algorithm

The weighting adaptation algorithm $M_i(x(t))$ is another problem we have to face. From previous context, we know the fuzzy filter output is $y = \sum_{i=1}^{r} M_i(x(t)) * C_i$, where M_i is the firing result of input membership function $\mu_i^{mad\_in}$. Substitution of the expression for fuzzy number in eq. (13) to M_i gives

$$M_i = \mu_i \cdot w_i = \begin{cases} \frac{a_i - x}{\alpha_i} \cdot w_{ai}, \ a_i \le x \le m_i \\ \frac{x - b_i}{\beta_i} \cdot w_{bi}, \ m_i \le x \le \beta_i \end{cases} \quad (21)$$

where the case of $x = m_i$ is simplified by applying the equal sign to interval of membership limits. The w_i is defined as a firing index, which can check that if the input variable is within the interval of considering fuzzy number. So firing index can be determine by following rules:

If $a_i \le x \le m_i$ then $w_{ai} = 1, \ w_{bi} = 0$
If $m_i \le x \le b_i$ then $w_{ai} = 0, \ w_{bi} = 1$
Then the eq. (21) can be rewritten as

$$M_i = (a_i - x) \cdot w_{ai} / \alpha_i + (x - b_i) \cdot w_{bi} / \beta_i \quad (22)$$

Applying the above into eq. (17), we could obtain the fuzzy filter output. Before presenting the final filter notation, we have defined y_i as a fuzzy filter bank. $y_i = M_i \cdot C_i$ (23)
For combining the variable in equal sign right side, we create a firing coefficient vector
$\mathbf{W} = [\mathrm{w}_1 \ \mathrm{w}_2 \ ... \ \mathrm{w}_r]$; r is the fuzzy rule number.
If $a_i < x < m_i$, then $\mathrm{w}_i = [-k_{ia} \quad 0]^T$.
If $m_i < x < b_i$, then $\mathrm{w}_i = [0 \quad k_{ib}]^T$.
Otherwise $\mathrm{w}_i = [0 \quad 0]^T$
where k_{ia} and k_{ib} is the coefficient of filter bank.

$$k_{ia} = C_i / (\alpha_i - m_i) \text{ and } k_{ib} = C_i / (m_i - \beta_i) \quad (24)$$

In the final we get fuzzy filter bank as

$$y_i = [x - a_i \quad x - b_i]^T \cdot w_i \quad (25)$$

and the filter output is obtained as $Y = \sum_{i=1}^{r} y_i$ (26)

For solving the condition when input variable is out of the universal set, we must specify $x - a_{first} = a_{first} - m_{first}$ and $x - b_{last} = m_{last} - b_{last}$ respectively. The subscripts '*first*' and '*last*' mean the index indicated to first and last membership function.

4. Simulation result

In this example we illustrate how to transform Mamdani fuzzy model to T-S fuzzy model. Making use of the approximate T-S model, we can examine the fuzzy filter weighting. The comparison between output of original Mamdani and proposed fuzzy filter is also presented.

Table 1 The rule base of Mamdani fuzzy system

x	**NB**	**NS**	**ZE**	**PS**	**PB**
u	PB	PS	ZE	NS	NB

Firstly, we developed a single input and 5 rules Mamdani fuzzy system, the membership function and rule base is defined as follows:
The x is the fuzzy input variable and u is the fuzzy output. Then we need to create the Sugeno fuzzy model to approximate a Mamdani fuzzy system. From eq. (6) the Sugeno fuzzy system can be defined as: IF x is Px, then $y = C_i$

The membership function of original Mamdani fuzzy system is illustrated in Figure 4 and the rule base is displayed in Table 1. From eq. (13), the important parameter of L-R type Mamdani fuzzy system with triangular shape is described as follows.

$A = [a_1 \quad a_2 \quad \ldots \quad a_i]$=[10 7.5 2 0 -2]

$B = [b_1 \quad b_2 \quad \ldots \quad b_i]$=[2 0 -2 -7.5 -10]

m=[7.5 2 0 -2 -7.5]

Applying the center of gravity defuzzfication method d to evaluate the $\mu_i^{mad-out}$. We get

$$C_i = [-886 \quad -493 \quad 0 \quad 493 \quad 886].$$

And the filter weighting can also be achieved by (21) and (22)

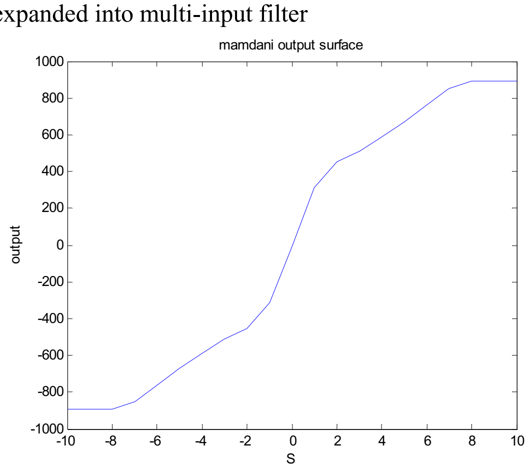
$$W = \begin{bmatrix} 354.4 & 89.63 & 0 & -246.5 & -161.09 \\ 161.09 & 246.5 & 0 & -89.63 & -354.4 \end{bmatrix}$$

Figure 5 show the comparison between Mamdani and T-S fuzzy filter. Figure 3(a) is the result of Mamdani output surface and Figure 3(b) is the result of TS-fuzzy output surface.

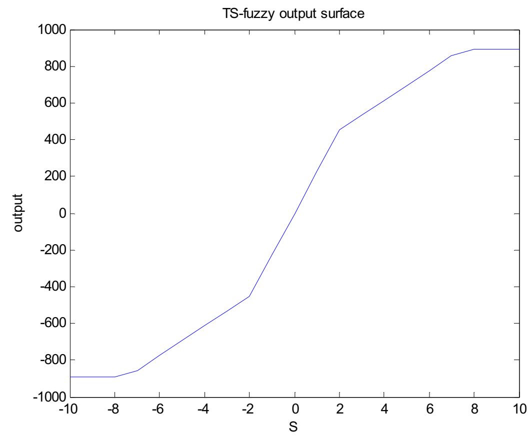
5. Conclusion

From pervious derivation, we have obtained a compact fuzzy filter form, which is suitable for real-time industrial applications. The simulation result demonstrates the similarity between T-S and mamdani fuzzy system. This shows that the performance of fuzzy filter is acceptable.

In the future, we will apply the proposed fuzzy filter in practical application. For example, the T-S fuzzy filter is suitable for implementation on FPGA. In addition, the single input fuzzy filter also has to be expanded into multi-input filter



(a) The result of Mamdani output surface



(b) The result of T-S fuzzy output surface

Figure 3 Comparison between original and approximate fuzzy filter

. References

[1] M. J. Er, B. H. Kee, and C. C. Tan, "Design and development of an intelligent controller for a pole-balancing robot," *Microprocessors and Microsystems*, vol. 26, no. 9-10, pp. 433-448, Dec.2002.

[2] T. H. S. Li, C. Shih-Jie, and C. Yi-Xiang, "Implementation of human-like driving skills by autonomous fuzzy behavior control on an FPGA-based car-like mobile robot," *Industrial Electronics, IEEE Transactions on*, vol. 50, no. 5, pp. 867-880, 2003.

[3] F. Russo, "Fuzzy systems in instrumentation: fuzzy signal processing," *Instrumentation and Measurement, IEEE Transactions on*, vol. 45, no. 2, pp. 683-689, 1996.

[4] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling," *Fuzzy Systems, IEEE Transactions on*, vol. 1, no. 1, p. 7, 1993.

[5] A. Costa, A. De Gloria, P. Faraboschi, A. Pagni, and G. Rizzotto, "Hardware solutions for fuzzy control," *Proceedings of the IEEE*, vol. 83, no. 3, pp. 422-434, 1995.

[6] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," *IEEE Trans. Syst. Man. Cyber.*, vol. 15, pp. 116-132, 1985.

[7] K. Tanaka and H. O. Wang, *Fuzzy control system dseign and analysis* . New York: A Wiley-Interscience Publication, 2001.

An Intelligent Quasi Fuzzy sliding Controller with Application to Seesaw Systems

Tsai-Cheng Li[1], Fang-Ming Yu[2], and Huan-Wen Wu[3]

[1]*St. John's & St. Mary's Institute of Technology Department of Mechanical Engineering*
li@mail.sjsmit.edu.tw

[2]*St. John's & St. Mary's Institute of Technology, Department of computer Science and Information Engineering*
[3]*Nan Jeon Institute of Technology Department of Mechanical Engineering*

Abstract

A new design technique using a single fuzzy input variable instead of the design method of the switching line to reduce the fuzzy controller rules for nonlinear systems is proposed. This technique only needs a single fuzzy input, and the deriving process is similar to the fuzzy sliding mode control (FSMC), called the single-input quasi fuzzy sliding mode control (SQ-FSMC). The advantage of the present approach is that the total number of rules is greatly reduced compared with that of two-dimensional FSMC. In addition, generations and tuning of control rules are more tractable. Other attractive feature of the approach includes the insensitivity to the uncertainties and disturbances. Simulation results are presented to demonstrate these features of the present approach. Finally, the experiment of a seesaw system subject to external disturbance and load is given to show the effectiveness and feasibility of the approach.

Keywords: Fuzzy control, Sliding mode control, Signed distance, Nonlinear system

1. Introduction

In recent years, some results of the fuzzy sliding mode control (FSMC) [2] has been reported. To decrease the number of rules in the rule base, several authors have suggested using a composite state to obtain a fuzzy sliding mode controller described in the previous work. The advantage of such controllers is that the number of rules is reduced from m^n to m^2 in Hwang [2]. Since the FSMC combines both fuzzy control and sliding mode control, the performance of system is superior to that using only one control theory. The main idea of the FSMC is to use the reaching condition $s\dot{s} < 0$, and let s and $\dot{s}$ be fuzzy variables. The rule base is then constructed in a two dimensional space. As for the system, which is higher than the second order, the fuzzy rules number is expected to be much more. Hence, the number of fuzzy rules should be reduced.

The SMC exhibits high frequency oscillations in its output, causing a problem known as the chattering phenomena. Chattering is highly undesirable because it can excite the high frequency dynamics of the system. So, we apply the fuzzy control principle to overcome this drawback.

In this work, we will develop a new control method called single-input quasi fuzzy sliding mode control (SQ-FSMC), the deriving process is similar to the FSMC. The similar idea is also introduced in [5]. We suggest a new variable called the signed distance [3], which is the distance to the actual state from the hyperplane. The signed distance is the single fuzzy input variable of SQ-FSMC. Hence, the number of fuzzy rules is greatly reduced compared with that of the FSMC. Furthermore, the Lyapunov function is composed of the signed distance, and then the control input u can be easily obtained. In this way, the control system is completely robust and asymptotic stability is guaranteed even if the disturbance exists. Moreover, the simulation and experimental result can verify the efficacy of the proposed method.

The SQ-FSMC can handle this chattering problem effectively by adjusting the control input near the sliding hyperplane. Another advantages of the performance in SQ-FSMC are those there is no need of exact mathematical model, the rule reduction, and robustness. Application to a seesaw system is given to show the effectiveness and feasibility of the approach.

2. System description and fuzzy sliding mode control

In this section, systems are described, and the FSMC for the systems is reviewed [2].

2.1. System description

Consider an n^{th}-order nonlinear system described by the following state-space model in a canonical form:

$\dot{x}_1(t) = x_2(t)$,

$\dot{x}_2(t) = x_3(t)$,

$$\vdots$$
$$\dot{x}_n(t) = f(\boldsymbol{x}) + b(\boldsymbol{x})u + d(t) , \qquad (2.1)$$
and
$$y(t) = x_1(t) \text{ , for } t \geq 0 , \qquad (2.2)$$
where $\boldsymbol{x} = [x_1, x_2, \cdots, x_n]^T$ is the state vector, , u is the control input, $f(\boldsymbol{x})$ and $b(\boldsymbol{x})$ are nonlinear functions, $y(t)$ is the system output, and $d(t)$ represents the external disturbance. The superscript T stands for the transpose of matrix. If the reference input $y_r(t)$ is a step function, then the above dynamic equations can be transformed into the following state equations with error signal $e_1 = y_r - y$ and its derivatives as state variables:
$$\dot{e}_1(t) = e_2(t) ,$$
$$\dot{e}_2(t) = e_3(t) ,$$
$$\vdots$$
$$\dot{e}_n(t) = -f(\boldsymbol{e}) - b(\boldsymbol{e})u - d(t) \text{ , for } t \geq 0 \qquad (2.3)$$
Let $\boldsymbol{e} = [e_1, e_2, \cdots, e_n]^T \in R^n$ and $\delta \in R$. A linear functional $\boldsymbol{s} : \boldsymbol{e} \to \delta$ is defined by
$$s(\boldsymbol{e}) = \boldsymbol{ce} , \qquad (2.4)$$
where $\boldsymbol{c} = [c_1, c_2, \cdots, c_{n-1}, 1]$, $\boldsymbol{c} \in \mathrm{R}^n$. For simplicity, the following equation will be used for (2.4):
$$s = \boldsymbol{ce} . \qquad (2.5)$$
Then, a sliding hyperplane can be represented as $s=0$ or $s = c_1 e_1 + c_2 e_2 + \cdots + e_n = 0$. (2.6)
The control purpose is to force the trajectory $\boldsymbol{e}$ to reach the sliding surface s in finite time and eventually sliding toward the origin $\boldsymbol{e} = \boldsymbol{0}$.

2.2. Fuzzy sliding mode control

As described in the last subsection, there are several methods to choose vector $\boldsymbol{c}$ [2], [4]. In this subsection, we briefly review the FSMC method. Based on the sliding mode control principle, the objective is to design a control law u such that the reaching condition:
$$s\dot{s} < 0 , \qquad (2.7)$$
is satisfied, where $\dot{s}$ represents the time derivative of s. The FSMC is shown in Fig. 1, where s and $\dot{s}$ are the inputs of the FSMC and u denotes the output of the FSMC.

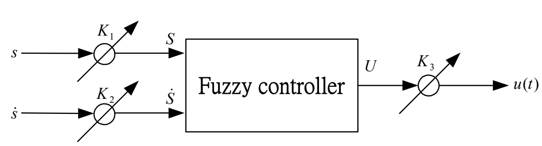


Fig. 1. The block diagram of the FSMC.

All the universes of $S, \dot{S}$ and U are arranged from –1 to 1. Thus, the range of nonfuzzy variables s, $\dot{s}$ and u must be scaled to fit the universe of fuzzified variable S, $\dot{S}$ and U with scaling factors K1, K2 and K3 respectively.

$$S = \mathrm{K1} \cdot s(\mathrm{t})$$
$$\dot{S} = \mathrm{K2} \cdot \dot{s}(\mathrm{t})$$
$$u(t) = \mathrm{K3} \cdot U \qquad (2.8)$$
For simplicity, a triangular type membership function is chosen for the aforementioned fuzzy variables.

The principle of the fuzzy rules base for the FSMC is explained below. By taking the time derivative of both sides of (2.5), we obtain
$$\dot{s} = \sum_{i=1}^{n-1} c_i e_{i+1} - f(\boldsymbol{e}) - b(\boldsymbol{e})u - d(t) , \qquad (2.9)$$
Then, multiplying both sides of the above equation by s leads to
$$s\dot{s} = \sum_{i=1}^{n-1} c_i e_{i+1} s - f(\boldsymbol{e})s - b(\boldsymbol{e})us - d(t)s , \qquad (2.10)$$
Here, we assume that $b(\boldsymbol{e}) > 0$ for any $\boldsymbol{e}$. In (2.9), it is seen that $\dot{s}$ increases as u decreases and vice versa. Equation (2.10) implies that if $s > 0$, then increasing u will make $s\dot{s}$ decrease and that if $s < 0$, then decreasing u will make $s\dot{s}$ decrease.

Based on the above qualitative analysis, one can design the control input u in an attempt to satisfy the reaching condition $s\dot{s} < 0$.

3. Design of single-input quasi-FSMC

Before deriving the SQ-FSMC method, we introduce a new variable called the signed distance [3]. And the switching line for a second-order system is defined by:
$$s : \dot{e} + c_1 e = 0 . \qquad (3.1)$$
Let $A(e, \dot{e})$ be the intersection point of the switching line and the line perpendicular to the switching line from an operating point $B(e_1, \dot{e}_1)$, as illustrated in Fig. 2. Then d_1 , the distance between $A(e, \dot{e})$ and $B(e_1, \dot{e}_1)$, can be expressed by the following equation:
$$d_1 = [(e - e_1)^2 + (\dot{e} - \dot{e}_1)^2]^{1/2} = \frac{|\dot{e}_1 + c_1 e_1|}{\sqrt{1 + c_1^2}} \qquad (3.2)$$

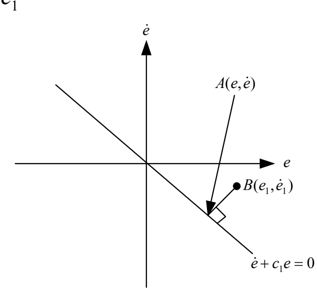


Fig. 2. Derivation of a signed distance

The signed distance d_s is defined for a general point $B(e, \dot{e})$ as follows:

$$d_s = \mathrm{sgn}(s)\frac{|\dot{e}+c_1 e|}{\sqrt{1+c_1^2}}$$

$$= \frac{\dot{e}+c_1 e}{\sqrt{1+c_1^2}} = \frac{s}{\sqrt{1+c_1^2}} . \tag{3.3}$$

where $$\mathrm{sgn}(s) = \begin{cases} 1 & \text{for } s>0 \\ -1 & \text{for } s<0 \end{cases} \tag{3.4}$$

The case can be extended to the n^{th}-order system. However, it is different from the switching hyperplane, i.e.,

$$s = e^{(n-1)} + c_{n-1}e^{n-2} + \cdots + c_2\dot{e} + c_1 e = 0 . \tag{3.5}$$

Hence, d_s is changed to be a general signed distance D_s as follows:

$$D_s = \frac{e^{(n-1)} + c_{n-1}e^{n-2} + \cdots + c_2\dot{e} + c_1 e}{\sqrt{1+c_{n-1}^2+\cdots+c_2^2+c_1^2}}$$

$$= \frac{s}{\sqrt{1+c_{n-1}^2+\cdots+c_2^2+c_1^2}} \tag{3.6}$$

The SQ-FSMC is shown in Fig. 3, where D_s and u are the input and output of the SQ-FSMC, respectively.

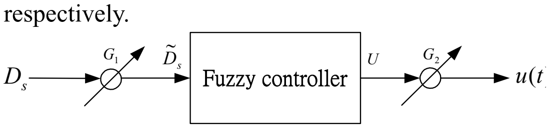


Fig. 3. The block diagram of the SQ-FSMC.

All the universes of D_s and U are arranged from –1 to 1. Thus, the range of nonfuzzy variables D_s and u must be scaled to fit the universe of fuzzified variable D_s and U with scaling factors G1 and G2, respectively, namely,

$$\tilde{D}_s = \mathrm{G1}\cdot D_s(t) , \quad u(t) = \mathrm{G2}\cdot U \tag{3.7}$$

In the implementation, a triangular type membership function is chosen for the aforementioned fuzzy variables. We apply the Mamdani's product-sum operation fuzzy implication and adopt the center of gravity as defuzzification method.

Next, we shall present the new control method. In subsection 2.2, among the relation between s, $\dot{s}$ and u. It is seen that s increases as u decreases and vice versa. Besides, if $s>0$, then increasing u will make $s\dot{s}$ decrease and that if $s<0$, then decreasing u will make $s\dot{s}$ increase. We then choose a Lyapunov function

$$V = \frac{1}{2}D_s^{\,2} . \tag{3.8}$$

Then $$\dot{V} = D_s\dot{D}_s = \frac{s\dot{s}}{1+c_{n-1}^2+\cdots+c_2^2+c_1^2} \tag{3.9}$$

Hence, it is seen that if $s<0$, then $D_S<0$, increasing u will make $s\dot{s}$ decrease so that $\dot{V}<0$ and that if $s<0$, then $D_S<0$, decreasing u will make $s\dot{s}$ decrease so that $\dot{V}<0$. From the above relation, we can ensure and guarantee that the system is asymptotically stable. From this relation, we can conclude that: $u \propto D_S$, (3.10)

Hence, the fuzzy rule table can be established on a one-dimensional space of D_S shown in Table 1 instead of a two-dimensional space of e and $\dot{e}$.

Table 1. Rule table for SQ-FSMC

D_s	NB	NM	NS	ZE	PS	PM	PB
u	NB	NM	NS	ZE	PS	PM	PB

4. Application to a seesaw system

In this section, application to a seesaw system is given to show the effectiveness and feasibility of the approach.

41. System Model

The balancing mechanism of the seesaw is shown in Fig. 4 [6]. The dynamical equation can be represented as

$$u + mg\sin\theta - B\dot{x} = m\ddot{x}$$

$$(Mg\sin\theta)\cdot r_2 + mg\sin(\theta+\phi)\cdot\sqrt{(x^2+r_1^2)} + ur_1 - \lambda\dot{\theta} = I\ddot{\theta}$$

(4-1)

where λ denotes the damping coefficient of the angle and I is the inertia of the wedge which is given below.

$$I = \frac{1}{2}\rho abc\left(\frac{a}{24}+\frac{b^2}{2}\right) \tag{4-2}$$

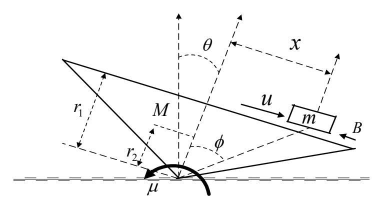


Fig. 4 The balancing mechanism of the inverted wedge

The Fig. 5 is the practical hardware structure of the seesaw system.

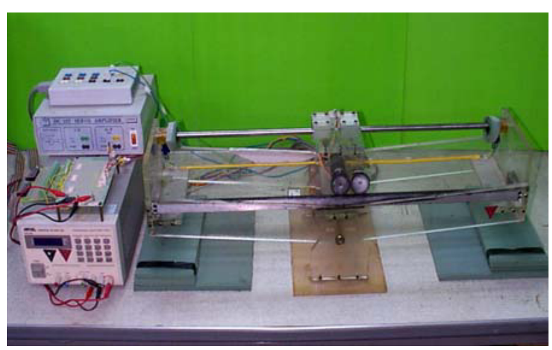


Fig. 5. The practical hardware structure of the seesaw system

4.2 The Application Results

For the seesaw system as in (5-1), the sliding surface that is represented by S has three variables as shown in Fig. 7, the variable of the angle θ between the center of gravity line and vertical line, the change of the angle $\dot{\theta}$ and the position of the cart from the origin x, respectively. Define the sliding surface as $S = c_1e_1 + c_2e_2 + c_3e_3$.

where $[e_1 \quad e_2 \quad e_3] = [0-\theta \quad 0-\dot{\theta} \quad 0-x]$. Due to the fact that the time derivative of state variable $\dot{x}$ affects the practical seesaw system a little, therefore, the chosen S is a function of only x, θ and $\dot{\theta}$.

Define a sliding surface $S = c_1e_1 + c_2e_2 + c_3e_3$, value $\boldsymbol{c}$ is chosen as:

$[c_1 \quad c_2 \quad c_3] = [1 \quad 1.6 \quad 5.1]$,and $[c_1 \quad c_2 \quad c_3] = [1 \quad 1.6 \quad 4.5]$ at loading (=1.5 kg). The scaling factors are chosen as G1=13, and G2=2 with or without loading (=1.5 kg). For the choice of coefficients c_1, c_2 and c_3, they are selected by trial-and-error and are determined under the condition of asymptotic stability, that is, with the same sign of coefficients and that if $t \to \infty$, then $\mathbf{x}(t) \to \mathbf{0}$.

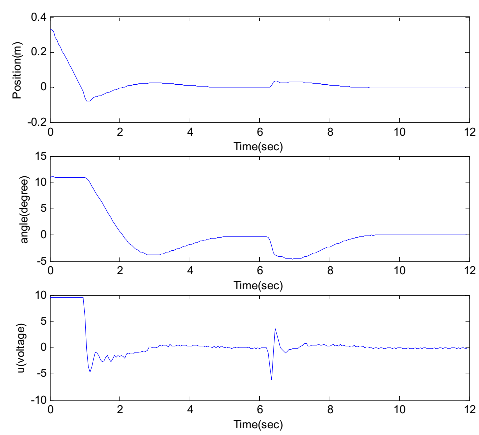


Fig. 6 SQ-FSMC control practical seesaw system

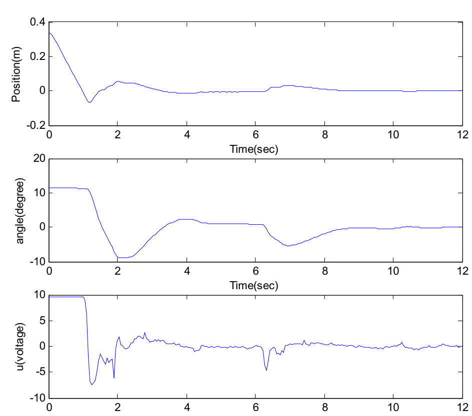


Fig. 7. SQ-FSMC control practical seesaw system with loading = 1.5 kg.

Based on what we want to balance the angle of inverted wedge and the position of cart, we can give the proper commands for balancing control with loadings, external disturbance, and thus construct the controller of the system. Under these circumstances, the experiment results for SQ-FSMC are shown in Fig. 6-7. It show the SQ-FSMC control response of the practical seesaw system with or without disturbance at loading = 1.5 kg.

6. Conclusions

We have shown that the SQ-FSMC developed in this paper can be easily applied to the n^{th} order single input single output nonlinear systems. The main features of the control method are summarized as follows:

(1) The fuzzy rules are greatly reduced. Some generations, modification, and tuning of control rules are more tractable.

(2) We may tune the control input u to make V become negative so as to ensure that the system is asymptotical stable.

(3) The chattering phenomena can also be dealt with very well without complicated mathematics.

(4) Experimental results for a seesaw system verify the efficacy of the proposed technique and its superiority to conventional sliding mode control in the senses of robustness .

In the future, the multi-input multi-output variable structure control in nonlinear systems will be investigated in depth.

References

[1]. E.H. Mamdani, " Applications of fuzzy algorithms for simple dynamic plants," Proc. IEE 121 1974, pp. 1585-1588.

[2]. Guang-Chyan Hwang, Shih-Chang Lin, " A stability approach to fuzzy control design for nonlinear systems," Fuzzy Sets and Systems vol. 48 1992, pp. 279-287.

[3]. Byung-Jae Choi, Seong-Woo Kwak, Byung Kook Kim, " Design of a single-input fuzzy logic controller and its properties," Fuzzy Sets and Systems vol. 106 1999, pp.299-308.

[4]. V. I. Utkin, K. D. Young, " Methods for constructing discontinuity planes in multidimensional variable structure systems, " Automation Remote Control, vol. 39 1979, pp. 1466-1470.

[5]. Shi-Yuan Chen, Fang-Ming Yu, and Hung-Yuan Chung, "Decoupled Fuzzy Controller Design with Single-Input Fuzzy Logic," Fuzzy Sets and Systems vol. 129, 2002, pp. 335-342.

[6]. M. L. Lee, H. Y. Chung, "Hierarchical Fuzzy Control with Applications to Seesaw Systems", M. S. thesis, Department of Electrical Engineering, National Central University, 2000.

Reviewers

A

Kamel ADI
Mohamed Ally

B

Arvind Bansal
Nadia Berthouze

C

Augusto Celentano
Kai H. Chang
Rwei-Ching Chang
Shi Kuo Chang
Shih-Fu Chang
Han-Chieh Chao
Ing-Ray Chen
Ko-Shung Chen
Kune-Yao Chen
Meng Chang Chen
Ruei-Xi Chen
Shu-Ching Chen
Tzung-Shi Chen
Yuh-Shyan Chen
Chuan-Feng Chiu
Yonghee Choi
Zied Choukair
Junaid Ahsenali Chowdary
Anthony Cox

D

Fadi P. Deek
Lawrence Y. Deng
Ramachandra A. Dixit
Jing Dong
David H. C. Du
Jean-Luc Dugelay

E

Larbi Esmahi

F

Filomena Ferrucci
George Furnas

G

Athula Ginige
Stuart Goose
Carsten Griwodz
William Grosky

H

Masahito Hirakawa
Peter Douglas Holt
Hui-huang Hsu
Teh-Sheng Huang
Jason C. Hung

J

Jong-Hann Jean
Qun Jin

K

Zenon Kulpa
James Kwok

L

Wei Lai
Dong-Liang Lee
Stefano Levaildi
Stefano Levialdi
Sheng-Tun Li
Wen-Syan Li
Mark Liao
Chang-Ching Lin
Fuhua Lin
Kuan-Cheng Lin
Shieh-Shing Lin
Chien-Tsai Liu
Jonathan Liu
YingQun Liu

M

Jian-Hua Ma
Qusay H. Mahmoud
Paolo Maresca
Kim Marriott
Rym Mili
Nikolay Mirenkov

O

Chikara Ohta
Vincent Oria
Ming Ouhyoung

P

Jen-Kuei Peng
Joseph J. Pfeiffer

R

Syed M. Rahman
Chotirat Ratanamahatana

S

Prasan Kumar Sahoo
Yoshitaka Shibata
Kuei-Ping Shih
Timothy K. Shih
Ronggong Song
Margaret-Anne Storey
Mahbubur Rahman Syed

T

Makoto Takizawa
Jin Tamura
Steven L. Tanimoto
Gustavo A. S. Torrellas
Genny Tortora
Maurizio Tucci

U

Kuniaki Uehara
Joseph E. Urban

V

Athena Vakali
Theodora Varvarigou
Son T. Vuong

W

Ching-Sheng Wang
Chun-Chia Wang
San-Yuan Wang
Guido Wirtz
Eric Wong

X

X. W. Xu

Y

David Yang
Sheng-Yuan Yang
Shi-Nine Yang
Rentaro Yoshioka

Z

Chi Zhang
Kang Zhang

Authors Index

S

T

V

W

Y

Z

DMS 2006 Call For Papers

The Twelfth International Conference on Distributed Multimedia Systems

Grand Canyon, USA
August 30 – September 1, 2006

Organized by
Knowledge Systems Institute Graduate School

SCOPE

The DMS conference is an international conference series, which covers a wide spectrum of technique discussions, demonstrations, and student program/paper contests in the fields of distributed multimedia computing. Started in 1994, the series of conference has been held at Hawaii, Hong Kong, Vancouver, Taipei, Aizu-Wakamatsu, Japan, San Francisco, Miami and Canada. This time, the conference will be held at the Grand Canyon Best Western Squire Inn, USA.

TOPICS

The theme of DMS'2006 is how to bridge the gap between communication and intelligence. DMS'2006 focuses on techniques, systems, applications, and theories in the fields of distributed multimedia computing. The conference organizers seek contributions of high quality papers, panels or tutorials, addressing various aspects of distributed multimedia systems and applications, for presentation at the conference and publication in the proceedings.

Topics of interest include, but are not limited to:

Audio and video compression
MPEG, Quicktime, Windows API standards
Image, video, audio content analysis and indexing/retrieval
Image, video and audio watermark
3D audio and video
Computer graphics and animation
Modeling and analysis of distributed multimedia systems
OS support for distributed multimedia systems
Distributed multimedia databases and computing
Multi-paradigmatic information retrieval
Multimedia human-computer interaction
Multimedia communications and network architectures
Mobile networks and mobile computing
Multimedia stream synchronization and QoS control/scheduling
Multimedia software engineering
Multimedia processors and ASIC
Multimedia technologies for people with disability
Intelligent multimedia computing
Intelligent multimodal interaction
Multi agents, mobile agents and their applications
Communication encryption
Security of distributed computer systems
Web servers and services
XML applications
Java, VRML and multimedia languages
Visual and multidimensional languages for multimedia applications
Multimedia digital libraries and mail systems
Multimedia applications for CSCW
Multimedia authoring tools and intelligent tutoring
Cultural heritage multimedia applications
Tele-conferencing, tele-medicine and tele-lecture
Virtual reality, distributed virtual environment, and their applications
Virtual school, virtual university, and virtual learning community
Distance learning methodology, tools and systems
E-commerce, e-education and e-entertainment

The use of prototypes and demonstration video for presentations is encouraged.

WORKSHOPS AND SPECIAL SESSIONS

This year, DMS'2006 will be held in conjunction with workshops, conferences and special sections. Papers submitted to workshops or special sessions are invited by the program committee chairs of the workshops/sections. This joint organization aims to collect research results from different perspectives. The following workshops and conferences are being planned:

International Workshop on Visual Languages and Computing
International Workshop on Mobile Systems, E-Commerce and Agent Technology
International Workshop on Distance Education Technology

Please contact the conference program co-chairs if you are interested in organizing a workshop or a special session. A one-page proposal with the following items is required:

Title of the workshop/special session
Name of the program committee chair(s)
A list of program committee members
E-mail address of the corresponding program committee chair
A brief description of the theme

Each special session should include at least 5 invited (or accepted) papers. Each workshop should include at least 2 special sessions (10 papers). Paper review process of the workshop/special session will be arranged by the individual program committee chair. It is the responsibility of the individual chairs to ensure the quality of papers in the workshop/special sessions. The schedule of workshops/special sessions will be the same as the main conference (See Important Dates below). Each individual program committee chair can set up the Web site of the workshop/special session. However papers will be printed in the same volume as the main conference.

SPECIAL ISSUE OF JOURNALS

The DMS conference is closely coordinated with the International Journal of Software Engineering and Knowledge Engineering, the Journal of Visual Languages and Computing and the International Journal of Distance Education Technologies. Usually the best ranked papers from the conference, after rigorous reviews, extensive revisions and further enhancements, will appear in one or more special issues of the above journals. Papers suggested for the special issue(s) will be reviewed by external reviewers following the standard procedure of review stipulated by the respective journal.

CONFERENCE SITE (HOTEL INFORMATION)

The DMS2006 Conference will be held at the Grand Canyon Best Western Squire Inn http://book.bestwestern.com/bestwestern/productInfo.do?propertyCode=03049 (South Rim). The hotel has graciously made available for these limited dates (8/29/2005 – 9/1/2005) to DMS2006 attendees a discount rate of $115 US dollars for single/double, $125 US dollars for triple, and $135 US dollars for quadruple occupancy per night.

INFORMATION FOR AUTHORS

Papers must be written in English. An electronic version (Postscript, PDF, or MS Word format) of the full paper should be submitted using the following URL: http://conf.ksi.edu/dms06/submit/SubmitPaper.php. Please use Internet Explorer as the browser. Manuscript must include a 200-word abstract and no more than 6 pages of IEEE double column text (include figures and references).

INFORMATION FOR REVIEWERS

Papers submitted to DMS'06 will be reviewed electronically. The users (webmaster, program chair, reviewers...) can login using the following URL: http://conf.ksi.edu/dms06/review/pass.php.

If you have any questions or run into problems, please send e-mail to: dms@ksi.edu.

DMS 2006 Conference Secretariat
Knowledge Systems Institute Graduate School
3420 Main Street
Skokie, IL 60076
USA
Tel: 847-679-3135
Fax: 847-679-3166
E-mail: dms@ksi.edu

If you cannot submit electronically, please send four copies of the complete manuscript to the above postal address.

IMPORTANT DATES

March 1, 2006	*Paper submission due*
April 30, 2006	*Notification of acceptance*
May 20, 2006	*Final manuscript due*
May 20, 2006	*Early conference registration due*