International Journal on Advances in Intelligent Systems

2009 vol. 2 nr. 1

The *International Journal On Advances in Intelligent Systems* is Published by IARIA.
ISSN: 1942-2679
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Responsibility for the contents rests upon the authors and not upon IARIA, nor on IARIA volunteers, staff, or contractors.

IARIA is the owner of the publication and of editorial aspects. IARIA reserves the right to update the content for quality improvements.

Abstracting is permitted with credit to the source. Libraries are permitted to photocopy or print, providing the reference is mentioned and that the resulting material is made available at no cost.

Reference should mention:

International Journal On Advances in Intelligent Systems, issn 1942-2679
vol. 2, no. 1, year 2009, http://www.iariajournals.org/intelligent_systems/

The copyright for each included paper belongs to the authors. Republishing of same material, by authors or persons or organizations, is not allowed. Reprint rights can be granted by IARIA or by the authors, and must include proper reference.

Reference to an article in the journal is as follows:

<Author list>, "<Article title>"
International Journal On Advances in Intelligent Systems, issn 1942-2679
vol. 2, no. 1, year 2009,<start page>:<end page> , http://www.iariajournals.org/intelligent_systems/

IARIA journals are made available for free, proving the appropriate references are made when their content is used.

Sponsored by IARIA
www.iaria.org

Copyright © 2009 IARIA

International Journal On Advances in Intelligent Systems
Volume 2, Number 1, 2009

Editor-in-Chief

Freimut Bodendorf, University of Erlangen-Nuernberg, Germany

Editorial Advisory Board

- Dominic Greenwood, Whitestein Technologies AG, Switzerland
- Josef Noll, UiO/UNIK, Norway
- Said Tazi, LAAS-CNRS, Universite Toulouse 1, France
- Radu Calinescu, Oxford University, UK
- Weilian Su, Naval Postgraduate School - Monterey, USA

Autonomus and Autonomic Systems

- Michael Bauer, The University of Western Ontario, Canada
- Radu Calinescu, Oxford University, UK
- Larbi Esmahi, Athabasca University, Canada
- Florin Gheorghe Filip, Romanian Academy, Romania
- Adam M. Gadomski, ENEA, Italy
- Alex Galis, University College London, UK
- Michael Grottke, University of Erlangen-Nuremberg, Germany
- Nhien-An Le-Khac, University College Dublin, Ireland
- Fidel Liberal Malaina, University of the Basque Country, Spain
- Jeff Riley, Hewlett-Packard Australia, Australia
- Rainer Unland, University of Duisburg-Essen, Germany

Advanced Computer Human Interactions

- Freimut Bodendorf, University of Erlangen-Nuernberg Germany
- Daniel L. Farkas, Cedars-Sinai Medical Center - Los Angeles, USA
- Janusz Kacprzyk, Polish Academy of Sciences, Poland
- Lorenzo Masia, Italian Institute of Technology (IIT) - Genova, Italy
- Antony Satyadas, IBM, USA

Advanced Information Processing Technologies

- Mirela Danubianu, "Stefan cel Mare" University of Suceava, Romania
- Kemal A. Delic, HP Co., USA
- Sorin Georgescu, Ericsson Research, Canada
- Josef Noll, UiO/UNIK, Sweden
- Liviu Panait, Google Inc., USA
- Kenji Saito, Keio University, Japan

- Thomas C. Schmidt, University of Applied Sciences – Hamburg, Germany
- Karolj Skala, Rudjer Bokovic Institute - Zagreb, Croatia
- Chieh-yih Wan, Intel Corporation, USA
- Hoo Chong Wei, Motorola Inc, Malaysia

Ubiquitous Systems and Technologies

- Matthias Bohmer, Munster University of Applied Sciences, Germany
- Dominic Greenwood, Whitestein Technologies AG, Switzerland
- Arthur Herzog, Technische Universitat Darmstadt, Germany
- Reinhard Klemm, Avaya Labs Research-Basking Ridge, USA
- Said Tazi, LAAS-CNRS, Universite Toulouse 1, France

Advanced Computing

- Dumitru Dan Burdescu, University of Craiova, Romania
- Simon G. Fabri, University of Malta – Msida, Malta
- Matthieu Geist, Supelec / ArcelorMittal, France
- Jameleddine Hassine, Cisco Systems, Inc., Canada
- Sascha Opletal, Universitat Stuttgart, Germany
- Flavio Oquendo, European University of Brittany - UBS/VALORIA, France
- Meikel Poess, Oracle, USA
- Said Tazi, LAAS-CNRS, Universite de Toulouse / Universite Toulouse1, France
- Antonios Tsourdos, Cranfield University/Defence Academy of the United Kingdom, UK

Centric Systems and Technologies

- Razvan Andonie, Central Washington University - Ellensburg, USA / Transylvania University of Brasov, Romania
- Kong Cheng, Telcordia Research, USA
- Vitaly Klyuev, University of Aizu, Japan
- Josef Noll, ConnectedLife@UNIK / UiO- Kjeller, Norway
- Willy Picard, The Poznan University of Economics, Poland
- Roman Y. Shtykh, Waseda University, Japan
- Weilian Su, Naval Postgraduate School - Monterey, USA

GeoInformation and Web Services

- Christophe Claramunt, Naval Academy Research Institute, France
- Wu Chou, Avaya Labs Fellow, AVAYA, USA
- Suzana Dragicevic, Simon Fraser University, Canada
- Dumitru Roman, Semantic Technology Institute Innsbruck, Austria
- Emmanuel Stefanakis, Harokopio University, Greece

Semantic Processing

- Marsal Gavalda, Nexidia Inc.-Atlanta, USA & CUIMPB-Barcelona, Spain

- Christian F. Hempelmann, RiverGlass Inc. - Champaign & Purdue University - West Lafayette, USA
- Josef Noll, ConnectedLife@UNIK / UiO- Kjeller, Norway
- Massimo Paolucci, DOCOMO Communications Laboratories Europe GmbH – Munich, Germany
- Tassilo Pellegrini, Semantic Web Company, Austria
- Antonio Maria Rinaldi, Universita di Napoli Federico II - Napoli Italy
- Dumitru Roman, University of Innsbruck, Austria
- Umberto Straccia, ISTI – CNR, Italy
- Rene Witte, Concordia University, Canada
- Peter Yeh, Accenture Technology Labs, USA
- Filip Zavoral, Charles University in Prague, Czech Republic

International Journal On Advances in Intelligent Systems
Volume 2, Number 1, 2009

Foreword

The first 2009 number of the International Journal On Advances in Intelligent Systems compiles a set of papers with major enhancements based on previously awarded publications. It brings together a set of articles that share a common link to intelligent systems. For this issue, nineteen contributions have been selected.

The majority of the articles deal with topics from the area of autonomy, learning, systems focused on the user, and decision support.

In the first article, Izumi Kohno et al. propose InfoCruise, and information navigation approach which uses a decision tree in order to disambiguate search results. The process makes use of user interaction to zoom in on the exact search concept. Debbie Richards et al. follow with an expertise recommendation system. The proposal adds relevance to recommender systems by taking into account the profile of the recommender who is gathering feedback.

In the third article, Jon G. Hall and Lucia Rapanotti look at the engineering problem of design assurance and present several aspects of problem oriented engineering.

The next three articles deal with autonomic computing. To this end, Radu Calinescu presents a reconfigurable service-oriented architecture using existing technologies, standards, and modeling techniques. In the next article, Michael Kleis et al. present a management approach for improvement in autonomic service control for Next Generation Networks. The software aspect of a gradual transition to autonomic computing is presented by Edin Arnautovic et al.

The next six articles in this journal issue deal with user management, user perception, and personalization. Edward Stehle et al. looks at assessing the user perceived utility of a VoIP system. From the same area, Petteri Pöyhönen et al. analyze access selection methods in relation to user perception in a multi-operator environment. Helena Suvinen and Pertti Saariluoma consider the usability of Wiki systems in the context of the psychological background of users. Manuel Goetz et al. present a project related to developing and using and digital education environment with lessons learned and conclusions for future work. For further understanding the user, more specifically as pedestrians, Junya Nakata et al. present an emulator for optimizing a pedestrian tracking system. To conclude the section on personalization, Hiroyuki Yamahara et al. propose context awareness in order to personalize service providing. The immediate application of such a system would be in a home.

The last group of articles deals with frameworks and decision support systems. Riina Maigre et al. describe experiments with large web services and their autonomic composition. Jian Chen et al. propose a gradual adaption model for information recommendation based on user behavior. Malohat Ibrohimovna and Sonia Heemstra de Groot follow with an article on group-oriented resource-sharing technologies. Bernhard Freudenthaler et al. examine computer aid to support structural assessment of bridges. Willi Richert and Riccardo Tornese present a learning framework meant to autonomously evolve based on learning reinforcements. Hasan Ibne Akram and Mario Hoffmann an identity management that is user centric applied in an ambient environment.

Last but not least, Michifumi Yoshioka et al. present intelligent electronic nose systems for fire detection. High accuracy is obtained with the presented approach.

We hope that the contents of this journal will add to your understanding of intelligent systems, and that you will be inspired to contribute to IARIA's conferences that include topics relevant to this journal.

Freimut Bodendorf, Editor-in-Chief
Petre Dini, IARIA Advisory Committees Board Chair

International Journal On Advances in Intelligent Systems
Volume 2, Number 1, 2009

CONTENTS

InfoCruise: Information Navigation Using a Focus Facet Based on Context **1 - 11**
Izumi Kohno, NEC Corporation, Japan
Yoji Miyazaki, NEC Corporation, Japan
Teruya Ikegami, NEC Corporation, Japan
Masaki Hara, NEC Corporation, Japan
Koji Kida, NEC Corporation, Japan

Expertise Recommendation: A triangulated approach **12 - 25**
Debbie Richards, Macquarie University, Australia
Meredith Taylor, Macquarie University, Australia
Peter Busch, Macquarie University, Australia

Assurance-driven design in Problem Oriented Engineering **26 - 37**
Jon G. Hall, The Open University, UK
Lucia Rapanotti, The Open University, UK

Reconfigurable Service-Oriented Architecture for Autonomic Computing **38 - 57**
Radu Calinescu, University of Oxford, UK

Autonomic Service Control In Next Generation Networks **58 - 73**
Michael Kleis, Technische Universität München, Germany
Andreas Klenk, Fraunhofer FOKUS, Germany
Benoit Radier, France Télécom R&D, France
Sanaa Elmoumouhi, France Télécom R&D, France
Georg Carle, Fraunhofer FOKUS, Germany
Mikael Salaun, France Télécom R&D, France

High-level Models of Software-management Interactions and Tasks for Gradual Transition Towards Autonomic Computing **74 - 91**
Edin Arnautovic, Vienna University of Technology, Austria
Hermann Kaindl, Vienna University of Technology, Austria
Jürgen Falb, Vienna University of Technology, Austria
Roman Popp, Vienna University of Technology, Austria

Perception of Utility in Autonomic VoIP Systems **92 - 106**
Edward Stehle, Drexel University, USA
Maxim Shevertalov, Drexel University, USA
Paul deGrandis, Drexel University, USA
Spiros Mancoridis, Drexel University, USA

Moshe Kam, Drexel University, USA

Analysis of Enhanced Access Selection Methods and End-User Perception in Multi-operator Environment 107 - 125
Petteri Poyhonen, Nokia Siemens Networks, Finland
Jan Markendahl, Royal Institute of Technology, Sweden
Ove Strandberg, Nokia Siemens Networks, Finland
Janne Tuononen, Nokia Siemens Networks, Finland
Martin Johnsson, Ericsson AB, Sweden

Working Globally via Wikis while Innovating and Acting Together: Case Wiki-Based Knowledge Sharing Portal 126 - 138
Helena Suvinen, University of Jyväskylä, Finland
Pertti Saariluoma, University of Jyväskylä, Finland

Online Teaching and Learning – Developing and Using an eEducation Environment 139 - 151
Manuel Goetz, University of Bayreuth, Germany
Stefan Jablonski, University of Bayreuth, Germany
Michael Igler, University of Bayreuth, Germany
Stephanie Meerkamm, University of Bayreuth, Germany
Matthias Ehmann, University of Bayreuth, Germany

Distributed Emulator for Developing and Optimizing a Pedestrian Tracking System Using Active Tags 152 - 161
Junya Nakata, National Institute of Information and Communications Technology, Japan
Razvan Beuran, National Institute of Information and Communications Technology, Japan
Tetsuya Kawakami, Panasonic Corporation, Japan
Takashi Okada, Japan Advanced Institute of Science and Technology, Japan
Ken-ichi Chinen, Japan Advanced Institute of Science and Technology, Japan
Yasuo Tan, Japan Advanced Institute of Science and Technology, Japan
Yoichi Shinoda, Japan Advanced Institute of Science and Technology, Japan

A Home Context-Aware System with a Mechanism for Personalization of Service Providing 162 - 180
Hiroyuki Yamahara, Ritsumeikan University, Japan
Takanori Soma, Ritsumeikan University, Japan
Fumiko Harada, Ritsumeikan University, Japan
Hideyuki Takada, Ritsumeikan University, Japan
Yukihiro Shimada, GOV Co., Ltd., Japan
Hiromitsu Shimakawa, Ritsumeikan University, Japan

Dynamic Service Synthesis on a Large Service Models of a Federated Governmental Information System 181 - 191
Riina Maigre, Tallinn University of Technology, Estonia

Peep Küngas, SOA Trader, Ltd., Estonia
Mihhail Matskin, Royal Institute of Technology - KTH, Sweden // Norwegian University of Science and Technology - NTNU, Norway
Enn Tyugu, Tallinn University of Technology, Estonia

Gradual Adaption Model for Information Recommendation Based on User Access Behavior 192 - 202
Jian Chen, Waseda University, Japan
Roman Y. Shtykh, Waseda University, Japan
Qun Jin, Waseda University, Japan

Resources Sharing and Access Control in Group-oriented Networks: Fednet and Related Paradigms 203 - 225
Malohat Ibrohimovna, Technical University of Delft, The Netherlands
Sonia Heemstra de Groot, Technical University of Delft, The Netherlands

Case-based Decision Support for the Assessment of Bridges 226 - 240
Bernhard Freudenthaler, Johannes Kepler University, Austria
Reinhard Stumptner, Johannes Kepler University, Austria
Josef Küng, Johannes Kepler University, Austria
Georg Gutenbrunner, VCE Holding GmbH Vienna Consulting Engineers, Austria

ESLAS – a robust layered learning framework 241 - 253
Willi Richert, University of Paderborn, Germany
Riccardo Tornese, Politecnico di Milano, Italy

User-centric Identity Management in Ambient Environments 254 - 267
Hasan Ibne Akram, Fraunhofer Institute for Secure Information Technology, Germany
Mario Hoffmann, Fraunhofer Institute for Secure Information Technology, Germany

Intelligent Electronic Nose Systems with Metal Oxide Gas Sensors for Fire Detection 268 - 277
Michifumi Yoshioka, Osaka Prefecture University, Japan
Toru Fujinaka, Osaka Prefecture University, Japan
Sigeru Omatu, Osaka Prefecture University, Japan

InfoCruise: Information Navigation Using a Focus Facet Based on Context

Izumi Kohno Yoji Miyazaki Teruya Ikegami Masaki Hara Koji Kida
NEC Corporation
8916-47, Takayama-Cho, Ikoma, Nara, 630-0101, Japan
{kohno@ay, y-miyazaki@bc, t-ikegami@ct, m-hara@cw, kida@da}.jp.nec.com

Abstract

To find desired information by mobile phones, it is necessary to set search keywords easily and to explore a search if a user's information needs are not well defined. We propose an information navigation method to help users successfully find information by mobile phones. Our proposed method presents users a focus facet by analyzing the context about contents, users, and dialogs. The focus facet is presented each time users refine a search. They can select a keyword from the facet, check the search results, and then define their ambiguous needs through interaction. We compared our method with traditional search methods. InfoCruise effectively discovers desired information, because its discovery rate and the satisfaction rating are higher than those of the traditional search methods. InfoCruise efficiently searches, because both the keyword correction rate and the number of times users changed facets in it are lower than those in traditional search methods.

Keywords information retrieval, exploratory search, user interface, faceted navigation, user context

1. Introduction

With the spread of the Internet, a great deal of information about products or shops has become available online that people can locate by search systems, not only by PCs but also by mobile phones. For example, when a user wants to go out to eat, he might search for an appropriate restaurant by mobile phone. But several problems exist with mobile searches. First, since the user's search purpose is often ambiguous, he can't search well. When searching for a restaurant, many users don't know its name or fail to stumble upon the right search word. The second problem is the constraints of mobile devices. Users have difficulty viewing long lists of search results on the small screens of mobile phones. They might also feel annoyed by repeatedly setting and changing search keywords because a mobile phone's communication speed is low.

Faceted navigation is a solution when a user's search purpose is ambiguous [2], [3], [4], [5]. This search method uses the metadata of information. Metadata have several facets, which are attributes in various orthogonal sets of categories. Faceted navigation displays the aspects of a current results set, so users can switch easily between searching and browsing. It also only shows the populated facets when users drill down search results by facet. So, users can explore search results without dead-ends. Faceted navigation is a good search method when a user's search purpose is ambiguous, because users can combine querying and browsing to clarify their purposes and find information. Most faceted navigation systems show all facets and many facet values on a PC screen. They need a wide screen area. But since mobile phones have small screens, showing all facets, such as current faceted navigation systems do, seems difficult.

As a solution, we propose InfoCruise, an information navigation method to help users successfully find information by mobile phones [1]. We support searches on small screens and when the search purpose is ambiguous. Our proposed method calculates the priority of facets by analyzing the context about contents, users, and dialogs and presents keywords in a focus facet each time a search is refined.

The contents of this paper are organized as follows. Section 2 describes the kinds of contexts as factors that affect the priority of facets. Section 3 describes our navigation method's overview, and Section 4 describes our application system to search for restaurants. Section 5 describes a user study in which we evaluated our method's effectiveness and efficiency. Section 6 discusses the results of a user study, Section 7 describes related works, and Section 8 provides a conclusion.

2. Focus facet determined by context

Information has many attributes, and leveraging metadata for searching has received attention in recent years. Hearst et al. [4], [6] used metadata (facets and facet values) as a tool to guide users in formulating queries. Their interface uses metadata in a manner that allows users to both refine and expand the query. Users select some facets and decide facet values for their demand to search. But it seems difficult to show many facets on a mobile phone with a small screen, so deciding on a focus facet among all facets is crucial. We propose to select and present a focus facet based on the user context. The following three kinds of contexts as factors affect the priority of facets: contents, users, and dialogs.

2.1. Context about contents

The context about contents affects which facet will be focused on. One reason why formulating queries is difficult is a lack of knowledge about target information [7]. It is important for users to know characteristics of target information. We think characteristics of target information are the content distribution about various data attributes. For example, when a user finds a restaurant, which facet is important among style, price, or location depends on the content distribution of these attributes. If most of the restaurants around a certain location are Chinese (he might be in a Chinatown), the user should learn that Chinese is the major type of restaurant nearby. In that case, the user would rather search by the "style" facet first than by another facet.

We calculate the priority of facets using the content distribution of each facet.

2.2. Context about users

The context about the circumstances around users also affects which facet will be focused on. For example, when a user wants to find a restaurant around him, the "location" facet is important. If a user wants to find a restaurant at midnight, he would search for restaurants using "open time" first. When a user wants to find a restaurant while he is driving, he would search using "parking" first. If a user can search for restaurants with the facet that suits the user's context, he can find restaurants efficiently.

We prioritize the facet associated with user context. If we relate the "location" facet with the current position and the system detects a user's current position, the system prioritizes the "location" facet. We can relate the "open time" facet with the current time and the "parking" facet with the means of transportation.

2.3. Context about dialog

Context about dialog history also affects which facet will be focused on. If a user can't find the desired information on the first search, using a different facet from the first search might prove successful. For example, if a user can't find a good "Chinese" or "Japanese" restaurant using a "style" facet, she might have success using another facet such as "atmosphere" because she can change her aspect. We believe that context about dialog history, which means when a facet was used, is important for searching.

Instead of prioritizing the facet just used, we prioritize another appropriate facet. However, the priority of a facet used before gradually increases each time a user refines a search.

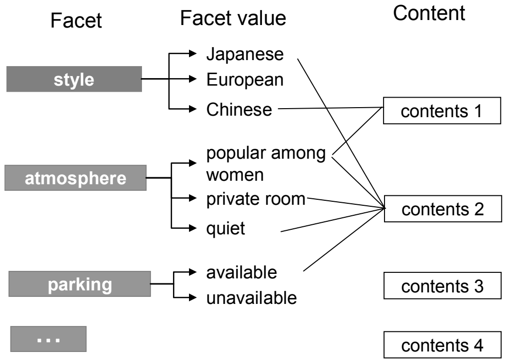


Fig. 1 Information structure

3. Navigation architecture overview

InfoCruise handles the information structure when contents have values in various orthogonal sets of categories. Fig. 1 shows an example where a restaurant has facet values in style, atmosphere, and parking facets. InfoCruise selects a focus facet among all facets and shows facet values as search keywords about the targeted information. InfoCruise also shows a dialog that explains why the facet and facet values were chosen.

Figure 2 shows the navigation architecture of InfoCruise. The system generates search keywords and dialogs in the following steps. The first step calculates the priority of facets by analyzing the context of the contents, users, and dialogs. In particular, the evaluated value of each facet is calculated using three

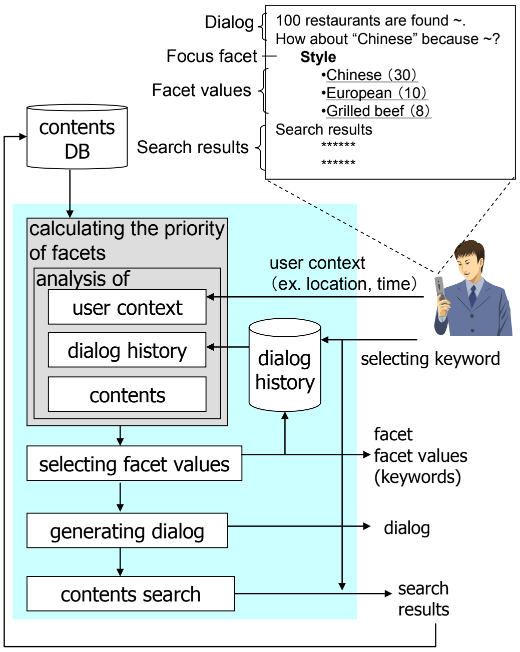


Fig. 2 Navigation architecture

contexts, and all facets are ranked by the evaluated values. The rule for the calculation of the priority of facets is defined as the "search strategy." The second step selects facet values from the highest priority facet. The third step generates a dialog.

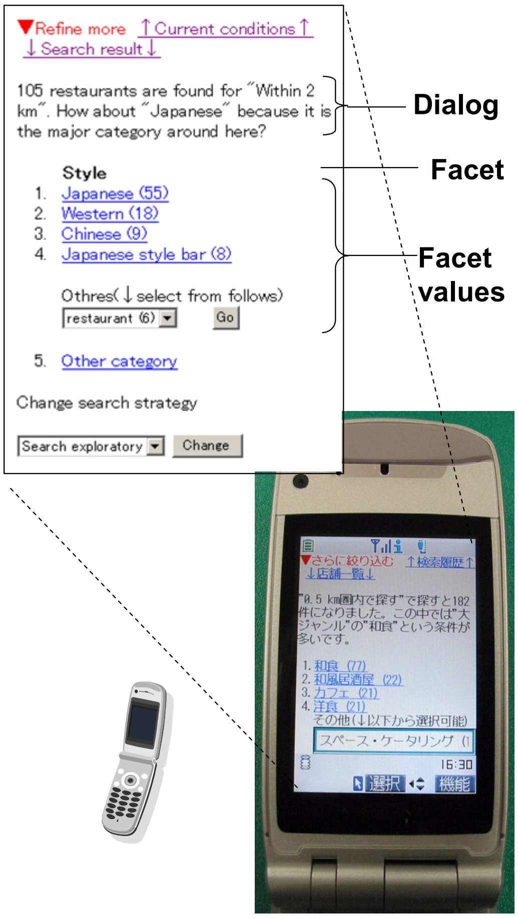
When a user selects a search keyword, the system refines the contents using that search keyword and generates new search keywords and dialogs for the search results once again. The contexts about the contents are updated using the refined content distribution of each facet, and the contexts about the dialogs are updated using the most recent history.

InfoCruise iteratively generates a focus facet and facet values as search keywords for the targeted information. Users can select desired search keywords, check the search results, and then define any ambiguous needs through interaction.

Our navigation architecture can define plural search strategies for some situations in an application. If users feel that the proposed search keywords aren't suitable, they can change the search strategy and select other search keywords.

4. Application system by mobile phone

To search for restaurants, we developed an application system composed of a server and a client. The client is a mobile phone. Search keywords and dialogs are created on the server computer and sent to a mobile phone. Fig. 3 is an image of InfoCruise on a mobile phone.


▼Refine more ↑Current conditions↑
↓Search result↓

105 restaurants are found for "Within 2 km". How about "Japanese" because it is the major category around here?

Style
1. Japanese (55)
2. Western (18)
3. Chinese (9)
4. Japanese style bar (8)

Othres(↓select from follows)
restaurant (6) ▾ Go

5. Other category

Change search strategy

Search exploratory ▾ Change

▼さらに絞り込む ↑検索履歴↑
↓店舗一覧↓

"0.5 km圏内で探す"で探すと182件になりました。この中では"大ジャンル"の"和食"という条件が多いです。

1. 和食 (77)
2. 和風居酒屋 (22)
3. カフェ (21)
4. 洋食 (21)
その他(↓以下から選択可能)
スペース・ケータリング (1

16:30
選択 機能

Fig. 3 Image of InfoCruise on a mobile phone

We used information about 4,500 restaurants in the Kansai area in Japan. The content has nine facets: location and open time, parking, price, interior design, service, cuisine, atmosphere, style, and sub-style. The facets and facet values were manually assigned. We also defined four user contexts: current position, current time, whether the user drives, and budget. These contexts will be acquired automatically in the future when the GPS function and electric money in mobile phones become widespread. However, we set these contexts manually in our prototype.

We prepared two search strategies for the retrieval of restaurants by mobile phones. One is an exploratory search strategy used when a user wants to know the most common type of restaurant in the area. In this strategy, we prioritize the facet with the largest difference among facet values because this is the most relevant information. Users select this strategy when they don’t know the area around them very well.

The second strategy is a quick search strategy used when a user wants to decide on a restaurant as soon as possible. In this strategy, we prioritize the facet in which the number of contents for each facet value is the most equal to refine the contents with certainty.

The advantage of the exploratory search strategy is that users can discover new and unexpected information by traversing the information set. But a disadvantage of the exploratory search strategy is that searches are time-consuming. In contrast, the advantage of the quick search strategy is that users can search rapidly. However users can’t extract new information using the quick search strategy.

We describe the method for ranking the facets of each search strategy as follows.

4.1 “Exploratory search” strategy

In the “exploratory search” strategy, the evaluated value of each facet is calculated by Formula (1) and the facet of the highest evaluated value is selected:

$$E_i = C_i \times U_i \times D_i \quad (1)$$

E_i is the evaluated value of facet i. C_i is the evaluated value of the context about the contents of facet i. U_i is the evaluated value of the context about the users of facet i. D_i is the evaluated value of the context about the dialog of facet i.

The evaluated value of the context about the contents, C_i, is calculated using the difference of the contents distribution among the facet values. If there is the largest difference among facet values in a facet, we think the facet has some characteristic that should be pointed out to users. C_i is calculated by formula (2):

$$C_i = \sum_j^{m_i} \left(N_{i,\max} - N_{i,j}\right)^2 \Big/ m_i \qquad (2)$$

$$N_{i,\max} = \max\left(N_{i,j}\right)$$

In Formula (2), m_i is the number of facet values in facet i. $N_{i,j}$ is the number of contents of facet value j in facet i, which is normalized by the summation of the contents of all facet values in facet i. For example in Fig. 4(A), value C_{style} is higher than value $C_{atmosphere}$, because there is greater variation in restaurant styles.

The evaluated value of the context about users, U_i, is defined as follows. We prioritize the facet associated with user context. If a user’s current position and time are set, the evaluated value of the “location and open time” facet is prioritized. If the user is driving, the evaluated value of the “parking” facet is prioritized. If a price range is set, the evaluated value of the “price” facet is prioritized. Specifically, $U_{location\ and\ open\ tim}$ are 1.8, $U_{parking}$ is 1.5, U_{price} is 1.3, and U_{others}=1.

The evaluated value of the context about dialog Di is calculated by Formula (3). We define the priority of a facet used just before lower than other facets, but its priority gradually increases each time a user refines a search:

$$D_i = \begin{cases} \alpha_i \times n_i & \alpha_i \times n_i < 1 \\ 1 & \alpha_i \times n_i \geq 1 \end{cases} \qquad (3)$$

In Formula (3), n_i is the number of dialogs since facet i was used (when facet i was used n_i=0), and α_i is the gradient. In this system α_i =0.01. We adjusted α_i so that a facet used before appears again after several dialogs.

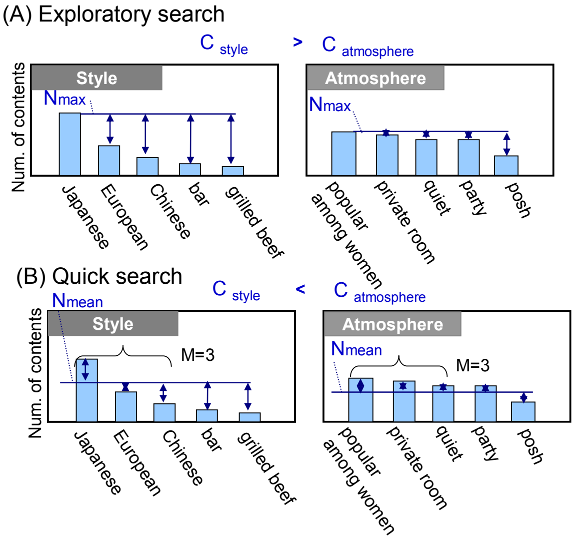


Fig. 4 Ranking based on content distribution

4.2 “Quick search” strategy

In the “quick search” strategy, the evaluated value of each facet is also calculated by formula (1) to select the facet of the highest evaluated value. The evaluated values of the context about dialog D_i, and users U_i are used in the same way as the “exploratory search.”

The evaluated value of the context about contents C_i is calculated using the evenness of content distribution. A user can refine the contents with certainty if he selects a facet in which the number of contents for each facet value is equal. The facet should be pointed out to users. C_i is calculated by Formula (4):

$$C_i = \exp(-\sum_k^M \left(N_{i,mean} - N_{i,k}\right)^2 \Big/ M) \qquad (4)$$

$$N_{i,mean} = \sum_k^M N_{i,k} / M$$

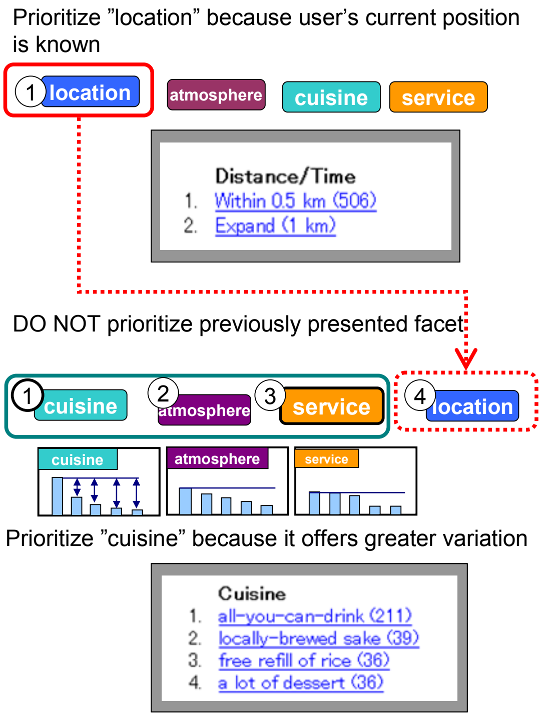
In Formula (4), M is determined by the number of search keywords that can be shown on a mobile phone. We calculate the variance of the top M facet values of each facet. C_i is raised when the variance becomes small. $N_{i,k}$ is the number of contents of facet value k in facet i, which is normalized by the summation of the contents of M facet values in facet i. For example in Fig. 4(B), value $C_{atmosphere}$ is higher than value C_{style}.

4.3 Interactive navigation

Ranking facets and selecting keywords are done each time a user refines a search. If a user's current position is known, the system prioritizes the "location" facet and presents search keywords about location to refine the contents, as shown in Fig. 5. When a user selects a search keyword in the "location" facet, its priority is decreased because it was just previously used. Next, the system calculates content distribution and prioritizes the facet with greater variation such as "cuisine" under the "exploratory search" strategy.

4. 4 Example of operation

We now explain an example of our system operation. In Fig. 6(A), a user sets his current position, budget, date, and time as user contexts. He also selects his favorite search strategy from "search exploratory" or "search quickly." The concept and purpose of each search strategy were explained to users in advance. His current position and time are set, so the evaluated value of the "location and open time" facet is prioritized. In Fig. 6(B), the system presents such dialogs to users as "If you select 'within 2 km' of Kyoto, 105 restaurants are available" and search keywords about distance. The system automatically searches for restaurants open at the current time and calculates the distance from the current position to each shop. The system defines the presentation number of shops and distance such as "105" restaurants and "2" km. When a user selects a search keyword "within 2 km," the system generates a focus facet and facet values once again to refine 105 contents. In Fig. 6(C), the system presents to a user a focus facet "style" with the largest difference among facet values under the search exploratory strategy. The system presents a dialog such as "105 restaurants are found within 2 km. How about Japanese because it is the most common type around here?" When a user selects "Japanese" as a search keyword, the system presents a focus facet "cuisine" with the largest difference among facet values to refine the 55 contents. Sometimes after a user selects keywords and the number of contents are reduced well (e.g., under five contents), the system stops proposing search keywords, as shown in Fig. 6(E).


Prioritize "location" because user's current position is known
1 location
atmosphere
cuisine
service
DO NOT prioritize previously presented facet
1 cuisine
2 atmosphere
3 service
4 location
cuisine
atmosphere
service
Prioritize "cuisine" because it offers greater variation

Distance/Time
1. Within 0.5 km (506)
2. Expand (1 km)

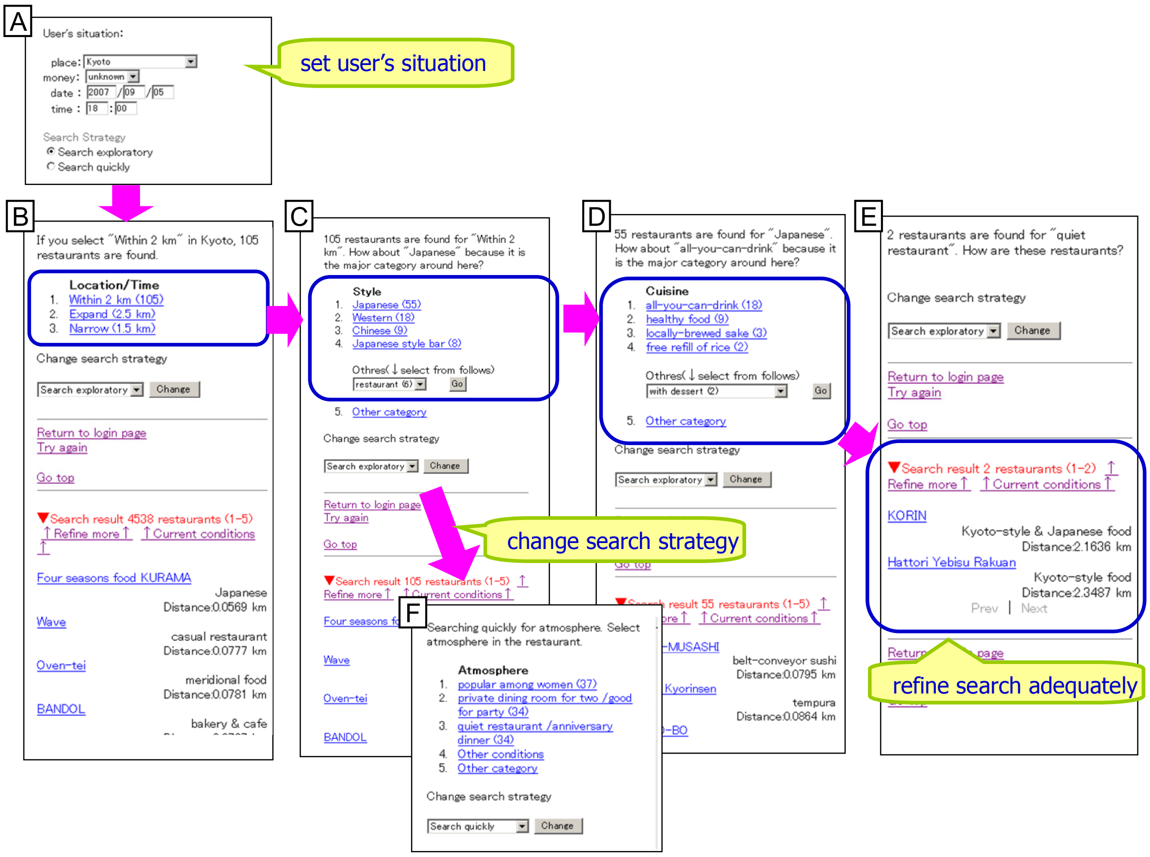
Cuisine
1. all-you-can-drink (211)
2. locally-brewed sake (39)
3. free refill of rice (36)
4. a lot of dessert (36)

Fig. 5 Ranking a facet iteratively

If the user feels that the proposed search keywords aren't suitable, she can change the search strategy, for example, from an exploratory to a quick search strategy (Fig. 6(C)). The system newly generates a focus facet and facet values to refine the same 105 contents and presents an "atmosphere" focus facet in which the number of contents for each facet value is the most equal. In Fig. 6(F), the system presents search keywords about atmosphere.

5. User study

We evaluated our information navigation method from the viewpoints of effectiveness and efficiency. We compared InfoCruise to two experimental systems using traditional search methods. One system searches by category (facet) using a pull down menu. All categories and search conditions are shown fixedly at all times. The other resembles traditional faceted navigation systems in which only one facet is shown and the appearance order of the facets is fixed, for example alphabetically.



A: User's situation:
place: Kyoto
money: unknown
date : 2007 / 09 / 05
time : 18 : 00
Search Strategy
● Search exploratory
○ Search quickly

B: If you select "Within 2 km" in Kyoto, 105 restaurants are found.
Location/Time
1. Within 2 km (105)
2. Expand (2.5 km)
3. Narrow (1.5 km)
Change search strategy
Search exploratory
Return to login page
Try again
Go top
▼Search result 4538 restaurants (1-5) ↑Refine more↑ ↑Current conditions↑
Four seasons food KURAMA — Japanese, Distance:0.0569 km
Wave — casual restaurant, Distance:0.0777 km
Oven-tei — meridional food, Distance:0.0781 km
BANDOL — bakery & cafe

C: 105 restaurants are found for "Within 2 km". How about "Japanese" because it is the major category around here?
Style
1. Japanese (55)
2. Western (18)
3. Chinese (9)
4. Japanese style bar (8)
Othres(↓select from follows)
restaurant (6)
5. Other category
Change search strategy
Search exploratory
Return to login page
Try again
Go top
▼Search result 105 restaurants (1-5) ↑ Refine more↑ ↑Current conditions↑
Four seasons fo…
Wave
Oven-tei
BANDOL

D: 55 restaurants are found for "Japanese". How about "all-you-can-drink" because it is the major category around here?
Cuisine
1. all-you-can-drink (18)
2. healthy food (9)
3. locally-brewed sake (3)
4. free refill of rice (2)
Othres(↓select from follows)
with dessert (2)
5. Other category
Change search strategy
Search exploratory
▼Search result 55 restaurants (1-5) ↑ …ore↑ ↑Current conditions↑
…-MUSASHI — belt-conveyor sushi, Distance:0.0795 km
Kyorinsen — tempura, Distance:0.0864 km
…-BO

E: 2 restaurants are found for "quiet restaurant". How are these restaurants?
Change search strategy
Search exploratory
Return to login page
Try again
Go top
▼Search result 2 restaurants (1-2) ↑ Refine more↑ ↑Current conditions↑
KORIN — Kyoto-style & Japanese food, Distance:2.1636 km
Hattori Yebisu Rakuan — Kyoto-style food, Distance:2.3487 km
Prev | Next

F: Searching quickly for atmosphere. Select atmosphere in the restaurant.
Atmosphere
1. popular among women (37)
2. private dining room for two /good for party (34)
3. quiet restaurant /anniversary dinner (34)
4. Other conditions
5. Other category
Change search strategy
Search quickly

Fig. 6 Example of search flow

5.1. Measures

Some measures were defined to evaluate our system's effectiveness and efficiency.

5.1.1. Effectiveness. Since InfoCruise is an exploratory search system, it has difficulty defining what contents a user wants before a search. Since we don't know the correct answer, we can't use such traditional search measures as relevance or recall ratios. Therefore, we measured the discovery rate and the satisfaction rating to evaluate effectiveness. The discovery rate is the number of tasks in which users found the content versus the number of all tasks. It does not matter whether the content found by the user is what the user wanted before the search. The satisfaction rating is a subjective ranking in which a score of 5 is defined as satisfied and 1 as dissatisfied. If a user can't find a restaurant, the rating is 0.

5.1.2. Efficiency. InfoCruise is an exploratory search system in which users repeat a search and confirm its results until they get their desired information. They clarify their needs through a process, so reducing the total operation time is not important. We used two indexes to measure whether futile operations decreased. First, we measured the correction rate to evaluate the efficiency. InfoCruise shows search keywords and the number of search results for each keyword to refine the current contents to help users preview their search results. Since users can select appropriate keywords using InfoCruise, so they won't need to correct keywords during the search. The correction rate is the number of corrected search keywords versus the number of search keywords set in one search. Second, we measured the number of times users changed facets for InfoCruise compared to traditional faceted navigation methods. InfoCruise analyzed the facet priority using context. This method will probably reduce the times users changed facets.

5.2. Experiment

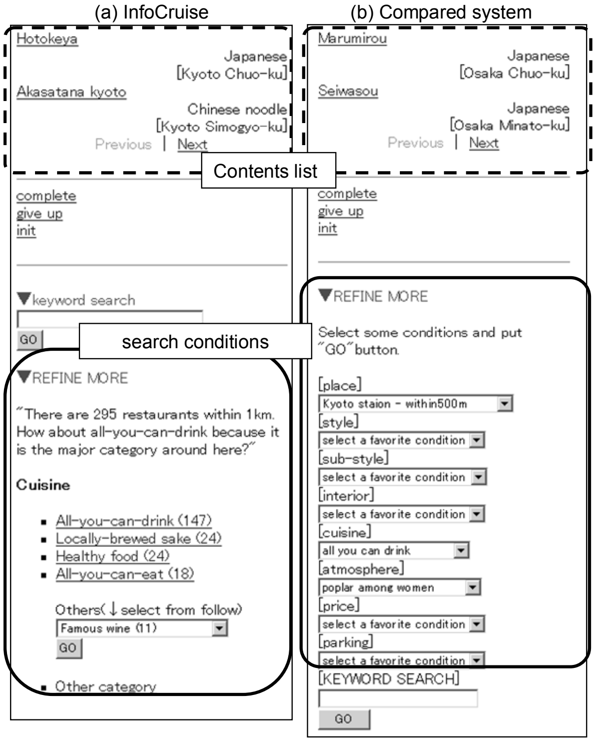
Two kinds of experiments were conduced to compare InfoCruise with traditional search systems.

5.2.1. Experiment 1: comparison with search by category system. We evaluated InfoCruise's effectiveness for a traditional search by a category system using the discovery rate and the satisfaction rating and its efficiency using the correction rate. The two systems had the same database and a similar UI (Fig. 7). The contents were located at the top of the screen and the search conditions at the bottom. The compared search system fixedly showed its categories and search conditions at all times. Users selected conditions using a pull-down menu and a "GO" button.

The 16 participants, who ranged in age from 20 to 40, commonly used mobile phones but they were not PC or Internet experts.

Participants searched for an appropriate restaurant under a designated situation. The tasks were prepared in five abstraction levels for search purposes to evaluate the InfoCruise's effectiveness when user needs were ambiguous. As shown in Fig. 8, each task defines a situation that designated some search conditions. Task T1 is the most abstract, which means the user needs were ambiguous. The situation is only designated by the location and date conditions. T5 is the most concrete, so its situation is designated by location, date, and two search conditions.

In our experiment, users executed five tasks (one for each task level) for InfoCruise and the compared system. Users pushed the "complete" button when they found a desired restaurant or users pushed the "give up" button when they wanted to quit. To measure the discovery rate, users could quit whenever they chose. After finishing one task, they completed a questionnaire containing satisfaction ratings and comments and then executed the next task. After finishing all five tasks of one experimental system, they moved to the other experimental system. The users executed five different tasks on the two experimental systems.



Hotokeya
Japanese
[Kyoto Chuo-ku]
Akasatana kyoto
Chinese noodle
[Kyoto Simogyo-ku]
Previous | Next

complete
give up
init

▼keyword search
GO

▼REFINE MORE

"There are 295 restaurants within 1km. How about all-you-can-drink because it is the major category around here?"

Cuisine

- All-you-can-drink (147)
- Locally-brewed sake (24)
- Healthy food (24)
- All-you-can-eat (18)

Others(↓select from follow)
Famous wine (11)
GO

- Other category

Marumirou
Japanese
[Osaka Chuo-ku]
Seiwasou
Japanese
[Osaka Minato-ku]
Previous | Next

complete
give up
init

▼REFINE MORE

Select some conditions and put "GO" button.

[place]
Kyoto staion - within500m
[style]
select a favorite condition
[sub-style]
select a favorite condition
[interior]
select a favorite condition
[cuisine]
all you can drink
[atmosphere]
poplar among women
[price]
select a favorite condition
[parking]
select a favorite condition
[KEYWORD SEARCH]
GO

Fig. 7 Interfaces of two experimental systems

Abstraction level

	Task level	Designated condition	Example (terms underlined here only)
abstract			Find your favorite restaurant under this situation.
	T1	Location, Date (near)	It is <u>9:30 pm</u> on <u>Friday</u> now. You are in <u>Osaka.</u>
	T2	Location, Date (distant)	You will go to <u>Kyoto</u> <u>at lunchtime</u> <u>next Sunday</u> .
	T3	Location, Date, Driving	It is <u>12:30 am</u> on <u>Sunday</u> now. You are driving in <u>Nara</u>.
concrete	T4	Location, Date, One search condition	You will go to <u>Osaka</u> <u>next Friday evening</u>. Your budget is <u>under 5000 yen</u>.
	T5	Location, date, Two search conditions	You will go to <u>Kobe</u> <u>next Saturday evening</u> <u>for a party.</u> Many people will come. There will be some <u>children</u>.

Fig. 8 Task examples of each abstraction level for search purposes

5.2.2. Experiment 2: comparison with faceted navigation system. We evaluated InfoCruise for a traditional search system like faceted navigation. InfoCruise presents user search keywords in a focus facet based on context. In contrast, the compared system shows search keywords in one facet whose appearance order is fixed alphabetically. If the ranking method of facet in InfoCruise is useful, the times that users change facets may be reduced. The main purpose of the experiment is to evaluate efficiency using the times users changed facets. We also measured the discovery rate, the satisfaction rating, and the correction rate for the traditional search system like faceted navigation.

The InfoCruise interfaces are shown in Fig7 (a). The interfaces of the compared system were the same as InfoCruise, but the appearance order of the facets was fixed alphabetically. If the users don't like the presented search conditions in one facet, they can change the facet by clicking the "Other category," in both InfoCruise and the compared system.

The procedures of Experiment 2 were the same as in Experiment 1, but only tasks 1, 3, and 5 were used: the five participants were in their 20s and 30s.

5.3. Results

We describe the results of the two experiments from the viewpoint of effectiveness and efficiency.

5.3.1. Effectiveness. Table 1 shows the results of Experiment 1 comparing InfoCruise to a traditional search system showing all categories any time fixedly. The discovery rate and the average satisfaction rating in all tasks and at each task level are shown in Table 1. The discovery rate of InfoCruise in all tasks (87%) was 10% higher than the compared system showing all categories any time (78%). The satisfaction rating of InfoCruise in all tasks (3.42) was 0.6 point higher than the compared system (2.8). The difference between the two systems was significant ($p<0.05$). The results suggest that InfoCruise is effective for desired information discovery.

Based on the discovery rate of each task level in Table 1, InfoCruise's ratings were better than the compared system in all task levels except Task 2. The difference between InfoCruise and the compared system in Task 1 was 25%, but the difference in Task 5 was only 4%. Furthermore, based on the satisfaction rating of each task level in Table 1, InfoCruise's rating was only significantly better than the compared system in T1. This means that InfoCruise is more effective under T1 than under T5. T1 is the most abstract task. The result suggests that InfoCruise is especially effective when user needs are ambiguous.

Table 3 shows the results of Experiment 2 comparing InfoCruise to a traditional search system like faceted navigation in which the appearance order of facets is fixed alphabetically. The discovery rate of InfoCruise in all tasks (100%) was higher than the compared system (87%). The satisfaction ratings between two systems were not different.

5.3.2. Efficiency. Table 2 shows the correction rate of Experiment 1 comparing InfoCruise to a traditional search system showing all categories any time. InfoCruise's correction rate, which was also calculated by each search strategy, was much smaller in all tasks (8.9%) than the compared system (37.4%). The correction rate of the "search exploratory" strategy was 5.9%, and the correction rate of the "search quickly" strategy was 16.7%. The average task completion time was 255 seconds for InfoCruise. In contrast, it was 291 seconds for the compared system. This means that since users can select appropriate keywords using InfoCruise, they don't need to correct keywords during the search and they can find contents more quickly than using the compared system.

Table 3 shows the number of times that users changed facets and the correction rate of Experiment 2 comparing InfoCruise to a traditional search system in which the appearance order of facets is fixed alphabetically. The number of changing facets for InfoCruise (0.8) was smaller than the compared system

Table 1 Discovery rate and satisfaction rating in Experiment 1

Discovery rate

Task level	T1	T2	T3	T4	T5	All tasks
InfoCruise	94%	88%	100%	81%	75%	87%
Compared	69%	94%	88%	64%	71%	78%

Satisfaction rating

Task level	T1	T2	T3	T4	T5	All tasks
InfoCruise	3.69	3.31	3.93	3.33	2.87	3.42
Compared	2.38	3.44	3.19	2.36	2.57	2.80
p	0.04	0.77	0.23	0.10	0.76	0.03

Table 2 Correction rate in Experiment 1

	Correction rate
InfoCruise	8.9%
Compared	37.4%

Search strategy	Correction rate	Num. of tasks
exploratory	5.9%	42
quick	16.7%	27
exploratory +quick	15.6%	6

Table 3 Times user changed facets, correction and discovery rates, and satisfaction rating in Experiment 2

	Num. of changing facets	Correction rate	Discovery rate	Satisfaction rating
InfoCruise	0.8	7.3%	100%	3.73
Compared	1.7	18.4%	87%	3.47

(1.7). The operation load for changing facets was reduced. InfoCruise's correction rate was also smaller than the compared system.

These two results suggest that users can search efficiently using InfoCruise.

5.3.3. User Comments. We gathered user comments by questionnaires. There were four possible answers to the question, "In what situation might you use InfoCruise?" The answers were: "when I want to quickly decide on a restaurant" (5 users), "when I don't have any idea where to go" (3), "when I want to pick a restaurant as close as possible" (1), and "when I have to decide on a restaurant in a new place" (1). Furthermore, there were four answers to the question, "What are the good points of InfoCruise?" The answers were: "I could find new information" (3), "it is easy to use on a mobile phone" (3), "I enjoyed the search process" (2), and "I got various kinds of information" (2).

6. Discussion

First, we discuss the effectiveness. InfoCruise's discovery rate was higher than the compared system showing all categories any time. Based on the user comments, users want to use InfoCruise when they don't have any idea where they will go or when they have to decide on a restaurant in a new place. In this situation, selecting keywords is difficult using such a traditional search method as the compared system because the user information needs are ambiguous. On the other hand, users can select keywords from presented keywords, define their needs through interaction, and find a restaurant using InfoCruise in this situation, so the discovery rate might increase.

InfoCruise's satisfaction rating was also higher than the compared system showing all categories any time. Based on user comments, they enjoyed finding new information and the search process itself. Even if the restaurant found was unexpected, the user might be satisfied with it through the process. On the other hand, when empty result sets were shown using the compared system, users felt like they had hit a dead end. After such a dead end, users seemed to settle on a kind of restaurant that they really didn't want. So the satisfaction rating of the compared system might be lower than InfoCruise.

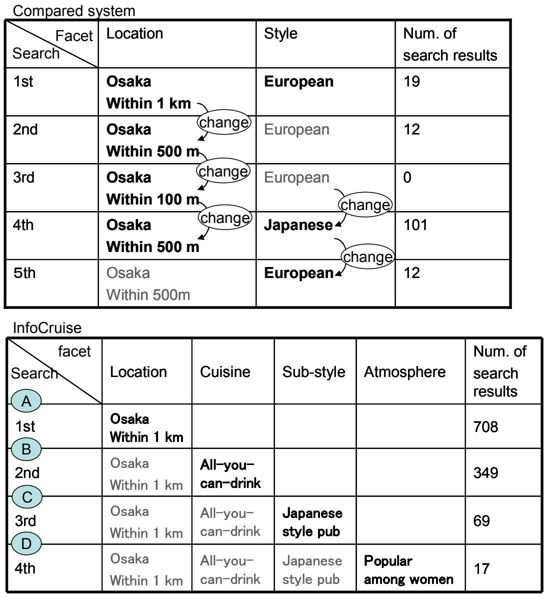
Next, we discuss the efficiency. The correction rate of InfoCruise was much smaller than the compared system showing all categories any time. Fig. 9 shows an example of a user's operation on both systems. The facets, the selected search conditions, and the number of search results are shown. In the compared system, a user changed the keyword set before on the same facet such as "location" and "style", perhaps because the number of search results was too great or too small. There was waste in the user operations, and the correction rate of the compared system was high. Otherwise, in InfoCruise, a user added keywords from different facets and gradually refined the contents. The user alternately selected a keyword from the location, cuisine, sub-style, and atmosphere facets. Fig. 10 shows the presented facet and search keywords in this experiment. The user's selection order followed the one presented by the system. Users can add keywords gradually after checking the search results, so the correction rate is low.

In addition, users can narrow down restaurants from an unexpected view with InfoCruise. For example in the experiment, one user, who selected a keyword in an atmosphere facet on the fourth search (Fig. 9), commented that "I don't have any restaurant preference." If the user hadn't searched for restaurants using InfoCruise, she wouldn't have selected a keyword from an atmosphere facet. The presented focus facet on InfoCruise inspired new interest in the user such as atmosphere.

Based on the correction rate of each search strategy, the correction rate for the "quick search" strategy was a little higher than for the "exploratory search" strategy. The search keywords presented for "quick search" might not fit the user's intention so well.

In the comparison between InfoCruise and the system in which the appearance order of facets is fixed alphabetically, the times users changed facets in InfoCruise was about half that of the compared system. So the algorithm for ranking facets might be useful to reduce user operation loads. If users aren't satisfied with the presented facet, they can change it. But in our experiment, many users tentatively selected a keyword from the facet presented by the system and corrected their selected keywords later. So the correction rate of the compared system was higher than InfoCruise.

In conclusion, since users enjoyed the search process and found a restaurant using InfoCruise even when their needs were ambiguous, both InfoCruise's discovery rate and satisfaction rating might increase. The result suggests that InfoCruise is effective for desired information discovery when a user's information needs are not well defined. In InfoCruise, a user added keywords from different facets and gradually refined the contents, so the keyword correction rate was low. The algorithm for ranking facets was also useful to reduce user operation loads. The result suggests that InfoCruise is efficient for searching for information.



Compared system

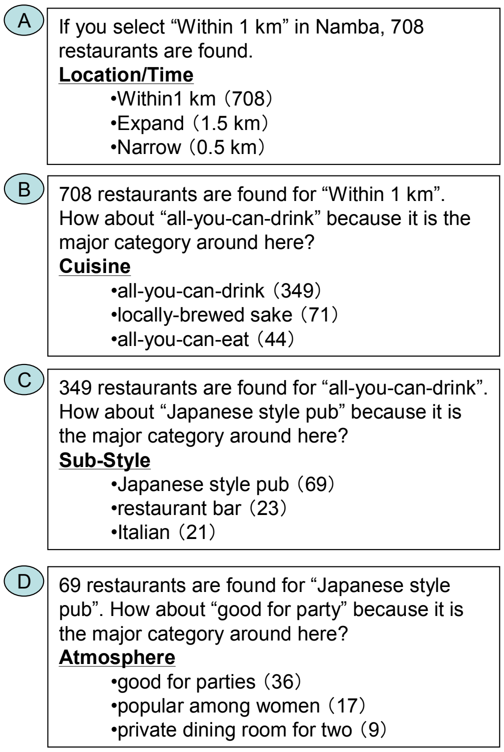
Search \ Facet	Location	Style	Num. of search results
1st	**Osaka** **Within 1 km**	**European**	19
2nd	**Osaka** **Within 500 m**	European	12
3rd	**Osaka** **Within 100 m**	European	0
4th	**Osaka** **Within 500 m**	**Japanese**	101
5th	Osaka Within 500m	**European**	12

change
change
change
change
change

InfoCruise

Search \ facet	Location	Cuisine	Sub-style	Atmosphere	Num. of search results
(A) 1st	**Osaka** **Within 1 km**				708
(B) 2nd	Osaka Within 1 km	**All-you-can-drink**			349
(C) 3rd	Osaka Within 1 km	All-you-can-drink	**Japanese style pub**		69
(D) 4th	Osaka Within 1 km	All-you-can-drink	Japanese style pub	**Popular among women**	17

Fig. 9 Example of user's operation in our experiment



A

If you select "Within 1 km" in Namba, 708 restaurants are found.

Location/Time

- •Within1 km（708）
- •Expand（1.5 km）
- •Narrow（0.5 km）

B

708 restaurants are found for "Within 1 km". How about "all-you-can-drink" because it is the major category around here?

Cuisine

- •all-you-can-drink（349）
- •locally-brewed sake（71）
- •all-you-can-eat（44）

C

349 restaurants are found for "all-you-can-drink". How about "Japanese style pub" because it is the major category around here?

Sub-Style

- •Japanese style pub（69）
- •restaurant bar（23）
- •Italian（21）

D

69 restaurants are found for "Japanese style pub". How about "good for party" because it is the major category around here?

Atmosphere

- •good for parties（36）
- •popular among women（17）
- •private dining room for two（9）

Fig. 10 Presenting facets and search keywords in the experiment

7. Related work

Our purpose is to support searches on small screens and when the user's search purpose is ambiguous. Faceted navigation is an existing solution for ambiguous search purposes. Most faceted navigation systems show all facets and many facet values on a screen [3], [4], [5]. If a screen is big enough such as a PC, these techniques are very effective for searches. However, mobile phones have small screens, and showing all facets such as existing faceted navigation systems seems difficult. Our method, which selects a focus facet for which users can choose a search keyword easily, is effective on the limited screens of mobile phones. FaThumb [2] is faceted navigation for mobile searches in which a number keypad navigates the metadata. Fathumb needs to make hierarchical metadata that include up to nine facets at one hierarchy. Our method sets the priority facets using context. If the number of facets increases, our method can automatically present a refined facet.

InfoCruise has two main features. One is presenting keywords to lighten the load for generating search queries. The other is interactive navigation. We discuss the relation between our method and other works by their features.

From the viewpoint of the presenting keyword function, Google Suggest, one technology that provides additional keywords after user input [8], presents keywords related to the input keyword by analyzing co-occurrence keywords selected from user logs. InfoCruise also presents keywords in a focus facet. But our method generates keywords based on the context about the users and contents. We can present keywords adapted to user circumstances or content distribution.

From the viewpoint of easing the load for generating search queries, research exists on query-free retrieval that automatically gathers information without user input. Reference [9] selects information based on text being written or read in Emacs. Reference [10] selects information based on the user's physical context such as his location, the people in the area, or the date. These systems automatically retrieve contents; otherwise InfoCruise presents keywords to refine them. We believe when systems infer using context, presenting keywords and confirming interactively might provide satisfying search results.

From the viewpoint of interactive navigation, relevance feedback [11] repeatedly performs a search and locates results close to a user's demand by generating queries from the search results users judged as matching their demands. The content of search results has many features, so the system doesn't know which feature is important from the user's evaluation of the content. A user selects a search keyword made directly from the metadata of the contents in InfoCruise. Our method could bring search results closer to a user's demand more directly.

8. Conclusion

We proposed InfoCruise, an information navigation method that presents keywords in a focus facet by analyzing the context about the contents, users, and dialogs. Our method calculates the priority of facets by analyzing three kinds of contexts. We developed an application system to search for restaurants by mobile phone and evaluated our method compared to traditional search methods. In our experiments, both the discovery rate and the satisfaction rating of our system are higher than those of the traditional search methods, so InfoCruise is effective to discover desired information. Both the keyword correction rate and the number of times users changed facets in our method are lower than those in the traditional search methods, so users can search efficiently with InfoCruise.

As future work, developing metadata assigned technology is important. In this paper, we used manually assigned metadata. Keywords must be automatically extracted that describe content features as well as people do. Moreover, keyword categorization technology is needed to make pairs of facets and facet values. In our method, dialog history affects the priority facets, but it is only a short-term history in one search. Long-term dialog history must also be treated because it reflects user preference. Our future work also involves calculating the priority facets based on context containing long-term dialog history.

References

[1] Izumi Kohno, Yoji Miyazaki, Masaki Hara, and Teruya Ikegami, InfoCruise: Information Navigation Presenting a Focus Facet Based on Context, IARIA, International Conference on Advances in Computer-Human Interaction, ACHI 2008, pp. 45-52, 2008.

[2] Amy K. Karlson, George G. Robertson, Daniel C. Robbins, Mary P. Czerwinski, and Greg R. Smith, FaThumb: a facet-based interface for mobile search, Proc. SIGCHI, ACM Press, pp. 711-720, 2006.

[3] Ka-Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst, Searching and organizing: Faceted metadata for image search and browsing, Proc. SIGCHI, ACM Press, pp. 401-408, 2003.

[4] Marti Hearst, Ame Elliot, Jennifer English, Rashimi Sinha, Kirsten Swearingen, and Ka-Ping Yee, Finding the flow in web site search, Comm. of the ACM, 45, 9, pp. 42-29, 2002.

[5] Endeca. http://endeca.com/

[6] Marti Hearst, Clustering versus faceted categories for information exploration, Comm. of the ACM, 49, 4, pp. 59-61, 2006.

[7] Ryen W. White, Bill Kules, Steven M. Drucker, and M. C. Schrafel, Supporting exploratory search, Comm. of the ACM, 49, 4, pp. 36-39, 2006.

[8] Google Suggest: http://www.google.com/

[9] Bradley Rhodes and Thad Starner, The Remembrance Agent: A continuously running automated information retrieval system, The Proceedings of The First International Conference on The Practical Application of Intelligent Agents and Multi Agent Technology (PAAM '96), pp. 487-495, 1996.

[10] Bradley Rhodes, Using Physical Context for Just-in-Time Information Retrieval, IEEE Transactions on Computers, Vol. 52, No. 8, pp. 1011-1014, 2003.

[11] Rocchio J., Relevance feedback in information retrieval: In the SMART Retrieval System, Experiments in Automatic Document Processing, pp. 313-323, 1971

Expertise Recommendation: A triangulated approach

Debbie Richards, Meredith Taylor, Peter Busch
Computing Department, Macquarie University, North Ryde, NSW, 2109, Australia
{richards,mtaylor,busch}@ics.mq.edu.au

***Abstract*— Recommender systems are becoming increasingly popular as a means for bringing products to the attention of online users. Similarly, they offer a means by which scarce resources in the form of human experts can be identified and accessed. However, if the information in the system is missing, incorrect or obsolete, recommendations will not be followed or even sought in the first place. Relying on individuals to validate and update this information is problematic. To provide automated acquisition and maintenance of information regarding who has expertise and in what areas, we employ data mining techniques. However, data mining will not provide the full picture and thus our outputs are reviewed by the experts themselves, providing a second means of validation. The third part to our triangulated approach is the use of profiles and the gathering of feedback from both searchers and experts to ensure that recommendations provided are satisfactory to both parties.**

Keywords: *expertise recommendation; recommender systems; data mining*

1. Introduction

Given the increasing recognition that an organization's most valuable resources are its people and the knowledge they hold, expertise location is becoming an important strategy to accessing and sharing that knowledge. In contrast to expert-systems, also known as knowledge-based systems, which seek to capture what it is that the expert knows so that it can be captured and reused, an expert/ise recommender system suggests who might know about what. The goal of the system is to point someone with a question to the person who has the appropriate knowledge. In the ideal situation the system provides a two-way communication channel connecting the knowledge holder and the knowledge seeker [1]. In some cases the inquirer's main interest is in the answer, in other cases the main interest is to find an expert who will handle the problem [2]. Knowledge about the expert's areas of expertise is needed for such a system. To discover this knowledge it is common to use data mining techniques [3]. Another alternative is to collect this information directly from the experts via self-reporting techniques (e.g. [4]). Individually both approaches have shortcomings.

Data mining relies on the existence of data which is, or is able to be, sufficiently structured to be used as input into one or more algorithms. This raises a number of issues: expertise could be identified from many different sources (e.g. publications, webpages, press releases, projects, grants, etc); these sources will vary across individuals and organizations; the format of these sources will vary across individuals and organization; most of these sources are free-format, unstructured and unclassified (a major hurdle if one wishes to use a supervised learning technique). Not only is the input to data mining an issue, each algorithm has its own strengths and limitations typically closely tied to the structure, amount and type of data and often dependent on the (type of) domain. Furthermore, the "knowledge learnt" tends to be restricted to types of output such as association rules, clusters or classification rules. Across domains, datasets and algorithms there is variation in the definition and identification of "interesting" concepts and validation of the output.

Due to these various limitations, an alternative to data mining and other automated searching techniques frequently used in recommender systems is the use of surveys/forms to be filled in by the domain expert which usually includes the selection of keywords relating to the individual's areas of expertise. The problem is then reduced to matching the searcher's query terms with the expert's keywords. This technique is often referred to as a yellow-pages approach to finding an expert as that is the way people usually find a plumber, lawyer or doctor. It is a simple and yet effective method for finding people who have certain skills. It works on the expectation that, for instance, only someone with legal training and qualifications will list themselves as a solicitor and that since they listed themselves, they are probably interested in receiving your call. Such assumptions are not always valid for recommender systems. The problems associated with the self-reporting approach include: experts failing to find the time to enter their data in the first place; data entered initially in a burst of enthusiasm by the individual or organization becoming obsolete or out-of-date; inaccurate and/or unvalidated self-reporting of expertise; and the levels of experience and degrees of currency are typically not being captured or maintained.

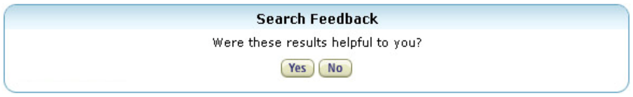
The approach we offer includes a combination of both techniques, together with a number of verification and validation techniques to improve the consistency, completeness and reliability of the knowledge, with the caveat that we cannot ensure consistency and completeness.

In the following section we consider related work on recommender systems. In section 3 we present two case studies conducted to elicit what is needed in an expertise recommender system. Section 4 presents the approach including results from a usability study performed to evaluate the prototype developed. Conclusions are given in Section 5.

2. Related work

Recommender systems share much in common with search engines which allow a user to enter keywords on a topic they are interested in and produce a list of links to resources based on those keywords and often also on the profile of the user. One of the most well known recommender systems is Amazon.com (http://www.amazon.com/) which allows the user to enter keywords and will search for books and other products based on those keywords. If the user does select one of the recommended products, the system will then suggest other products that it thinks the user might like based on the choices of other users who also selected the same product. That is, the system reasons that if a particular user likes the same product as 20 other users, they may also like some of the other products that those 20 users liked; this is known as collaborative filtering and is a common technique in modern recommender systems (e.g. [4]).

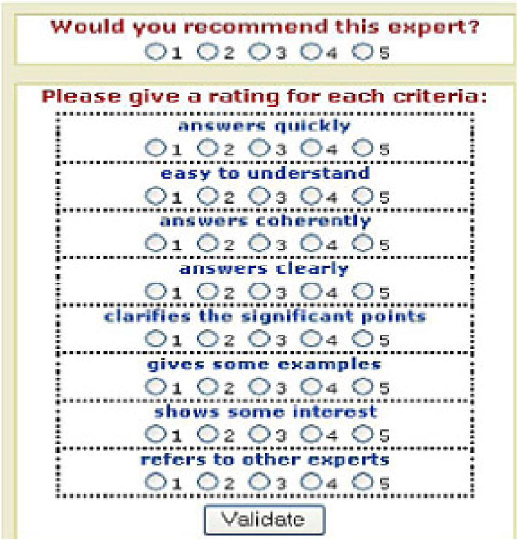
Recommender systems for experts work in much the same way in that they take a user's search query and try to find someone who has the expertise to answer that query. The user will then be given the expert's contact details. These systems are mainly used internally within organizations. Validation that the information provided was wanted and useful is missing in many systems. Amazon attempts to obtain feedback using the message box shown in Figure 1.


Search Feedback

Were these results helpful to you?

Figure.1. Search feedback from Amazon.com

Aïmeur et. al. [4] further explored the concept of validation of automatic identification of experts. They describe a recommender system called HELP which attempted to locate expertise and experts within an organization. The system included a database of questions asked previously by users and their respective answers. When a user searched for a solution to a particular problem, the question/answer database was first searched to see if their problem or a similar one had already been dealt with. If it had, the user was then presented with the solution, which they could choose to either accept or reject. If they rejected the solution, or if one wasn't found, the system would then search its user database for someone with the potential expertise to solve the problem. Users were thus required to register their own areas of expertise with the system. The rating system shown in Figure 2 served to somewhat validate the recommendation the user was given by storing the responses in the profile of the expert. If someone claimed to be an expert on a certain topic but consistently received low ratings, then they would not be recommended by the system if it was possible to recommend someone with a higher rating.


Would you recommend this expert?

○1 ○2 ○3 ○4 ○5

Please give a rating for each criteria:

answers quickly

○1 ○2 ○3 ○4 ○5

easy to understand

○1 ○2 ○3 ○4 ○5

answers coherently

○1 ○2 ○3 ○4 ○5

answers clearly

○1 ○2 ○3 ○4 ○5

clarifies the significant points

○1 ○2 ○3 ○4 ○5

gives some examples

○1 ○2 ○3 ○4 ○5

shows some interest

○1 ○2 ○3 ○4 ○5

refers to other experts

○1 ○2 ○3 ○4 ○5

Figure. 2: Evaluation of Experts [4]

An alternative to self-reporting recommender system, are fully automatic approaches to locate experts such as Who Knows [5] and SAGE [6] using inputs such email (e.g., Agent Amplified Communications [7]), bulletin boards (e.g. Contact Finder [8]), Web pages (e.g. YENTA [9] and MEMOIR [10]), program code (e.g. Expertise Recommender [11] and Expert Finder [12]), and technical reports (e.g. KCSR Expert Finder [13]). These techniques could be applied to the artifacts of social software systems (i.e. email, WebLogs and

Wikis) to provide automatic expert location. However, a review of these systems [3] found problems related to heterogeneous information sources, expertise analysis support, reusability and interoperability.

Quickstep, described in [14], is a recommender system for online papers. It uses unobtrusive monitoring of searchers' browsing behaviour to find what kind of papers a searcher is interested in and create an interest profile for each searcher. When a searcher searches for papers, only those papers that have not already been browsed by the searcher and have a topic of interest to the searcher are returned.

Feedback forms may be provided for a searcher to provide negative or positive comments on the recommendations after the item has been received by the searcher. One major problem with relying on the searchers to actively provide feedback as in [4] is that there is never a certainty that they will do so. Aïmeur *et al.* [4] reported that most people would provide positive ratings if any at all. On the other side, if a searcher had an extremely negative experience they may provide a rating, but otherwise may not bother. Even if the system sends regular emails to the searchers to remind them to provide feedback on a recommendation they were given, it is still not guaranteed that the searcher will do so. In fact, the searcher may regard the reminder emails as an annoyance and ignore them altogether.

Any recommender system that recommends items for purchase (such as Amazon.com, for example) can, to some extent, measure how valid and useful a recommendation was by recording if the searcher decided to buy the item that was recommended. However in most of the expert recommender systems encountered in the literature, there is no way of knowing whether a recommendation of an expert caused a searcher to contact the expert. These systems provide the names and contact details of recommended experts, but then leave it to the searcher to contact the experts at their discretion; thus they not only have no idea if a contacted expert was able to help the searcher, but also have no idea if a searcher even tried to contact the expert in the first place.

The lack of feedback problem is addressed in [4] by ensuring that all searcher and expert interaction is controlled by the system and by having profiles for both searchers and experts. However, their approach is geared towards providing quick solutions to problems rather than putting people in contact with one another. Thus it seems unrealistic to insist that all contact between a searcher and an expert be through the system, but it does make sense that searchers be allowed to make initial contact with experts via the system, and experts be allowed to send the initial response through the system so the system can gather data on an expert's availability and response time.

3. Exploratory case studies

As our goal is to create a recommender system that has the confidence and support of its intended users and goes beyond the yellow-pages model, it was essential that we not just review the literature but also analyse the experiences of practitioners frequently concerned with the task of locating experts and expertise. Thus we conducted two case studies.

3.1 Case study 1

Firstly we conducted interviews with seventeen personnel during the months of September and November 2006 within a Defence R & D organization which is expertise intensive. A series of questions was presented to our interviewees in an attempt to get them to present their experiences on accessing expertise in each of their fields (e.g. "What features or criteria do you use to determine if someone is an expert?", "How do you find what projects/problems people are working on?"). The questions were also designed to elicit barriers they faced to gaining expertise/finding an expert as well as assessing the quality of the expertise they were provided with. It is this last parameter (quality) that is the most difficult to assess.

The following five main questions were asked (with subquestions associated with each to further prompt answers):

1. How do you go about finding an expert? Please give one or two actual examples of how you have done this in the past.

- Have you used any tools or software support?
- What sources of information do you look at? (documents, email, websites, databases, personal reference)
- What role has the web played in finding information about expertise inside and external to your organisation?
- What mechanisms does the organization have to identify experts?
- What features or criteria do you use to determine if someone is an expert?
- Do you consider personal characteristics? If so, how would you work these out?
- What ranking/order would you give to the criteria that you use to identify/find a person?
- Does the importance of a criteria vary for different situations? If so, can you give examples from your

experience of when certain criteria were important and which were less important or unimportant in a different situation?
- How do you determine the person's level of expertise and the currency of that expertise?
- How do you find what projects/problems people are working on?
- Is it more useful to know what problem someone has been working on or the application/domain they worked within?
- How long do you usually have to find an expert? Do you need to find an answer to a problem as quick as possible, or is it in the project planning phase when you are looking for team members/mentors/advisors?

2. Do you use a different process depending on the location of the person, their status, the department they are in, or other factor? Please give one or more examples.

- Has your strategy changed over time?
- Does this process differ for people outside of your organisation?
- Who decides who is in a team? How do they decide? Please give an example of how a recent team you were involved with was put together.

3. What are the impediments or barriers to identifying an expert?

- What are the impediments or barriers to accessing an expert?
- What are the impediments or barriers to validating an expert?

4. Is there information you are interested in gaining access to but currently you are unaware of where you could find it or whether what exists is accessible or reliable?

- Do people advertise their skills? Should they advertise? What about bidding for projects?
- What role does budget, timeframe, resources play in finding an expert and then getting access to them?

5. How is trust developed regarding expertise in your organisation?

- How do you validate that someone is an expert and is that information passed on in some way to others in the organization?
- What mechanisms does the organization have to reward experts? Is there any incentive to be recognised as an expert or does this lead to more work or less time for your own work?

It was clear from our initial investigations in 2006, that a fully automated approach which advises who to contact, or a semi-automated approach using techniques such as SNA to answer "who knows who" would not deliver an optimal or widely accepted solution. There were basically two ways of accessing 'know-how' within the organization: 1) drawing upon one's social networks (a people-centric approach); 2) reading publications/project descriptions to determine who has the relevant expertise (a more algorithmic/automated type of approach). Issues that arose in accessing expert/ise included:

- Establishing trust and quantifying levels of trust. Interviewees using social networks to locate expertise mentioned the importance of trust between the members of the social network. Also mentioned was the difficulty in determining the level of truthfulness in journal and conference papers and how much trust should be placed in the reported results.
- The organisation experiences a high turnover of experts and thus a loss of expertise.
- Access to experts across organizational boundaries.
- Currency of expertise and the impact of currency on relevancy of expertise. Although the organisation attempted to maintain the currency of the expertise held by their experts by up-dating their knowledge and ensuring that the relevant training was administered, it was mentioned that knowing something is better than knowing nothing, even if the knowledge is not current.
- Short timeframes (hours/days) for decision making related to forming a new team and project.
- Information is typically classified and only available on a need to know basis.

In this organization the key concern was "how do you find someone to work on a particular project?" Using our interview transcripts we conducted content analysis as we had done with the literature. This revealed that people and projects, and to a lesser extent, tools and groups, play central roles in identifying who is an expert. The structure of the organization also played a major role in being able to find and access an expert.

The majority of the Defense R&D personnel possessed PhDs and had a high level of technical knowledge. We found that the senior personnel tended to place more importance on the quality of a potential expert's publications than less senior staff. All personnel interviewed mentioned the importance of social networking and maintaining contact with others in their field. Senior staff tended to have more international contacts as well as larger social networks within their organization. The most junior staff member we interviewed was the least concerned with finding experts and had the least number of contacts in other departments of the organization.

Three of the personnel interviewed were liaison librarians who were often required to do literature searches for staff members. They mentioned that staff members were starting to do their own web-based subject searches rather than asking the librarians for help. The librarians personally visited a number of staff members to keep them up to date with what is happening in the organization. They also told us that many staff members were socially isolated.

At the time of the interviews, the organization had no established or frequently updated expert recommender system. We were told that there had been attempts to create an expertise database, but it was difficult to keep up to date and was not current at the time of the interviews.

3.2 Case study 2

The second case study we conducted was within our own university, another expertise intensive organization. There are four key areas within the university concerned with locating and contacting experts. These are:

- the Research Office (RO) – concerned with finding experts for research projects;
- Development and External Relations Division (DERD) (now better known as Community Engagement) – concerned with finding experts for awards and for connecting Macquarie staff with industry and the community;
- Marketing and Public Relations (MPR) – concerned with finding experts for media interviews on behalf of journalists; and
- Teaching and Learning (TL) – concerned with finding experts for guest lectures and expertise related to teaching and learning such as skills and experience with working with groups or teams, giving iLectures, handling large classes, providing feedback, etc.

We interviewed representatives (the senior executive of the RO, director and two others in DERD, two senior people in MPR and the Chair of the University Learning and Teaching Committee for TL) from these four areas, asking them the same questions as in the first case study (removing questions that were specific to the first organization) as well as asking them their general requirements for an expert recommender system. Based on the findings from the literature and our own experiences, we were particularly interested in the answer to the following questions:

1. *If there was a system that recommends experts, would you be willing to spend a small amount of time providing feedback to the system to indicate whether a recommendation given to you was useful?*
2. *What sort of searcher feedback would be useful to you?*

A few of the representatives had reservations about the feedback form (Fig 2.) presented to them as an example of what feedback might look like. TL mentioned that any negative feedback on a person's teaching style would be taken very personally and is akin to criticising the person themselves, as teaching skills are not something that can be taught easily or learned right away. While it may be useful to indicate which experts have good and bad teaching skills, it may cause people to be more reluctant to use the system or volunteer to give guest lectures. MPR mentioned that journalists would be unlikely to fill out the feedback form unless they had a negative experience and it was immediately available.

The methods university representatives preferred with respect to finding experts were quite similar to those used by staff at the Defence R&D organisation. The RO searched through publication titles in the Integrated Research Information System (IRIS) to find an expert. MPR said that they frequently used Xpertnet, Macquarie University's expert recommender system for journalists, to locate experts.

The requirements listed by the representatives included:

- Searching by age (suggested by DERD for offering age-specific awards).
- Searching for expertise in general as well as expertise in a particular field (suggested by DERD for offering non-domain-specific awards).
- Details of well-established and/or recognised experts, such as professors, and lesser established, (e.g. Masters and PhD students) experts were needed, as well as non-academics performing relevant research. (This was suggested by DERD as they have found that people studying towards a higher degree were more likely to be interested in applying for an award).
- Details about an expert's teaching awards, publications, and grants, as well as information about the type of teaching skills they possess (e.g. if they have experience in student centered learning, or team teaching). (Suggested by TL as a method of judging the quality of the expert's teaching skills and finding out, for instance, if they would be suitable to give guest lectures or mentor new staff members).

- A model illustrating the relationships between experts (Suggested by RO to show joint grants and papers).
- Certain flags to restrict searches on grant applications that are confidential (suggested by RO).
- Experts who are registered in the system should be allowed to nominate availability (suggested by MPR as the response time of an expert was an issue).

4. Approach

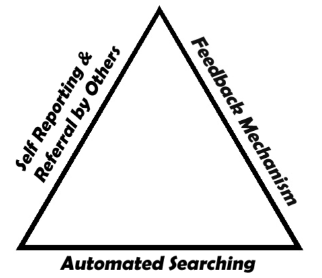


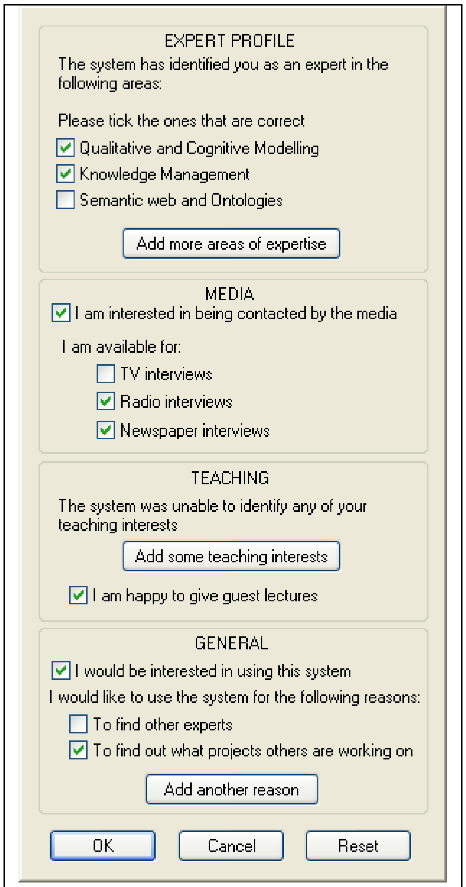
Figure.3. Our Triangulated approach

The case studies revealed that a combined automated and human-in-the-loop approach was necessary. As depicted in Figure 3, the proposed approach uses automated searching as a foundation from which the initial data is captured and against which the data is regularly cross checked. The key inputs to data mining include individual web pages, project/grant repositories, citation indexes (e.g. CiteSeer (http://citeseer.ist.psu.edu/) and publications databases. Within our university we will use the Integrated Research Information System (IRIS) as one of our data sources. IRIS consists of a collection of all publications and impact factors of individuals within the university. The latter is currently our most useful resource as it is the most structured.

In the information extraction technique we trialed [15], results were sent to each of the 20 identified experts in the Computing Department in an email giving them an opportunity to review and validate the areas of expertise found by the system. The result provided to each expert was a set of RFCD (Research Fields, Courses and Disciplines) codes as defined by the Australian Research Council (ARC). These codes were based on the expert's publications in IRIS and were used as an externally validated and publically recognized indicator of their research areas. In addition to validation of the outputs of datamining concerning their areas of expertise, we also propose that the expert would be able to provide additional information regarding their preferences as shown in Figure 4.

Using automated searching we can attach dates related to the expertise found to assist with currency. We can also keep statistics on the level of expertise using simple measures such as the number of publications in that area and Term Frequency Inverse Document Frequency (TFIDF).

This confirmation and correction by the expert of the results of automated searching provides a second dimension: *Self reporting and referral by others* which is based upon the first dimension: *Automated Searching*. By allowing experts to edit the automated results we are allowing them to self-report their areas of expertise, deleting or adding new areas as they see fit. The system would also allow for others, such as a PhD supervisor, to nominate or refer another person, such as their student.



EXPERT PROFILE

The system has identified you as an expert in the following areas:

Please tick the ones that are correct

☑ Qualitative and Cognitive Modelling
☑ Knowledge Management
□ Semantic web and Ontologies

Add more areas of expertise

MEDIA

☑ I am interested in being contacted by the media

I am available for:

□ TV interviews
☑ Radio interviews
☑ Newspaper interviews

TEACHING

The system was unable to identify any of your teaching interests

Add some teaching interests

☑ I am happy to give guest lectures

GENERAL

☑ I would be interested in using this system

I would like to use the system for the following reasons:

□ To find other experts
☑ To find out what projects others are working on

Add another reason

OK Cancel Reset

Figure 4. A simplified validation screen sent to the expert as a result of data mining from webpages, publication/citation databases, etc.

With many systems that rely entirely on self-reporting, some people will choose to simply not to have a profile rather than going to all the trouble of registering themselves and maintaining their profile. In large organizations it is also possible that some people may not even know about the system, especially if it is fairly new. To populate such a system it is most likely that some experts would need to be contacted either personally or via mass email and the rate at which experts are added to the system would be directly related to the rate at which experts (most of whom would be very busy) are able or willing to register themselves.

Rather than asking potential experts to "opt in", by using automated methods as the foundational first step in information acquisition, this system would instead ask them to "opt out" if they do not wish to be registered with the system. It would also be much easier for an expert to review a portfolio which has been automatically generated for them rather than have to format one for themselves. In addition to sharing the data validation and maintenance with the searcher, the system provides a standard way of describing its experts, which will simplify the searching process.

To address the issues of external validation, expertise currency and motivation to enter and maintain the data, individuals are sent the output from data mining at regular intervals, say twice yearly. Given the importance of reputation and track record in the university system we believe academics would be motivated to check what the system, as it reflects the data in the world about them, says and to correct any errors or omissions.

From discussions, interviews and personal experience the key question to be asked of the advice of any recommendation system is "was it useful?" As shown in Figure 3, the third support to our approach is a feedback mechanism that will allow the searchers of the system to validate the recommendations themselves.

Feedback will be gathered from both the searcher and the expert. As illustrated in the activity diagram in Figure 5, once an expert is recommended by the system the searcher can choose to contact the expert via a contact form provided by the system. The system would then email this message to the expert along with a link to a form where they can send their initial reply. In this way the expert can provide information about their availability and expertise as well as writing a personalized message to the searcher (Figure 6).

If the searcher wishes to contact the expert via another route, such as a phone call, the system would allow them to indicate this by clicking a link or button. If the searcher chooses this option, a message will be sent to the expert informing them that they may be contacted regarding the search terms entered by the searcher. The expert will be directed to a form where they can indicate if they are available and possess the expertise to address the query. This form would be similar to that in Fig 6 without the searcher's message at the top and the space to write a personalized message. The system will then be able to update the expert's statistics in the system and inform the searcher of the expert's status if the system has both searcher and expert profiles or if the searcher provides the system with their email address.

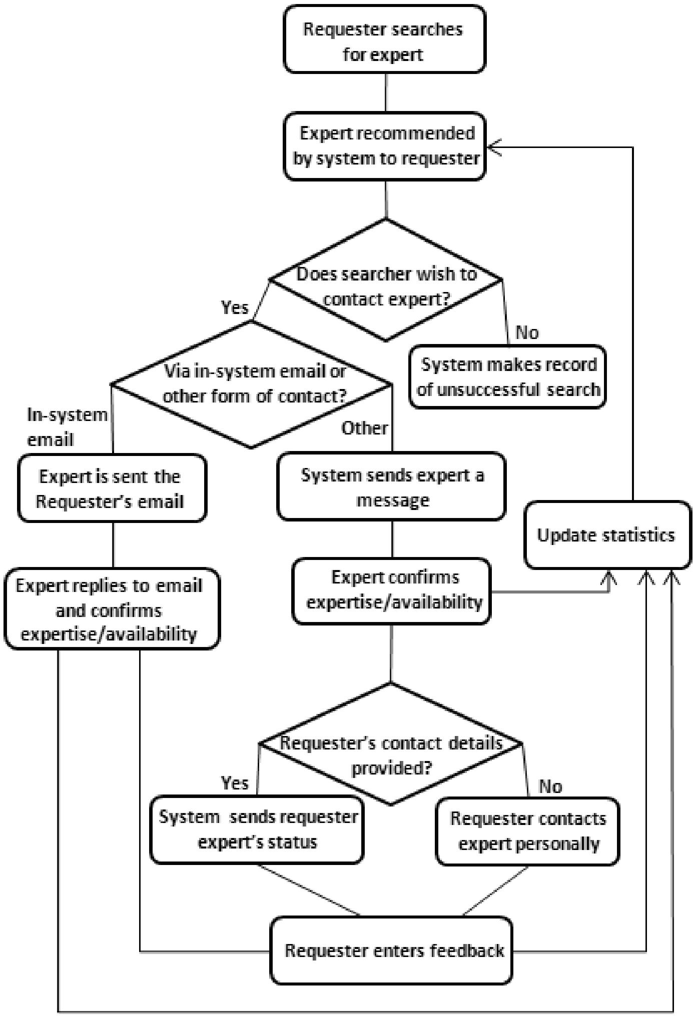
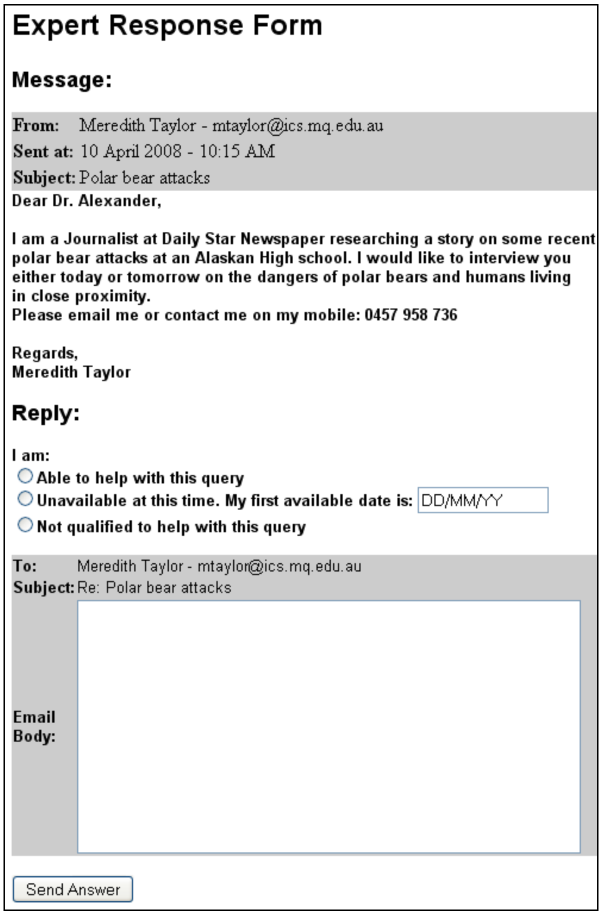

Requester searches for expert
Expert recommended by system to requester
Does searcher wish to contact expert?
Yes
No
System makes record of unsuccessful search
Via in-system email or other form of contact?
In-system email
Other
Expert is sent the Requester's email
System sends expert a message
Update statistics
Expert replies to email and confirms expertise/availability
Expert confirms expertise/availability
Requester's contact details provided?
Yes
No
System sends requester expert's status
Requester contacts expert personally
Requester enters feedback

Figure 5. Activity diagram depicting search flow between Searcher, Expert and System

Thus the system is able to keep track of what recommendations yield success (that is, which recommendations result in an expert being contacted). If the searcher does not indicate in any way that they wish to contact the expert, the system will store the search terms used for a brief period of time and observe if similar search terms also yield unsuccessful results. If so, the system will reevaluate the profiles that are recommended for those search terms.

After interacting with a recommended expert, a searcher may provide feedback on the expert whenever they wish. Otherwise a follow-up technique such as sending an email to the searcher a couple of weeks after the searcher makes contact with an expert could be used, reminding them to provide feedback.



Expert Response Form

Message:

From: Meredith Taylor - mtaylor@ics.mq.edu.au
Sent at: 10 April 2008 - 10:15 AM
Subject: Polar bear attacks

Dear Dr. Alexander,

I am a Journalist at Daily Star Newspaper researching a story on some recent polar bear attacks at an Alaskan High school. I would like to interview you either today or tomorrow on the dangers of polar bears and humans living in close proximity.
Please email me or contact me on my mobile: 0457 958 736

Regards,
Meredith Taylor

Reply:

I am:
○ Able to help with this query
○ Unavailable at this time. My first available date is: DD/MM/YY
○ Not qualified to help with this query

To: Meredith Taylor - mtaylor@ics.mq.edu.au
Subject: Re: Polar bear attacks
Email Body:

Figure 6. Example of an expert response form with searcher's message at the top.

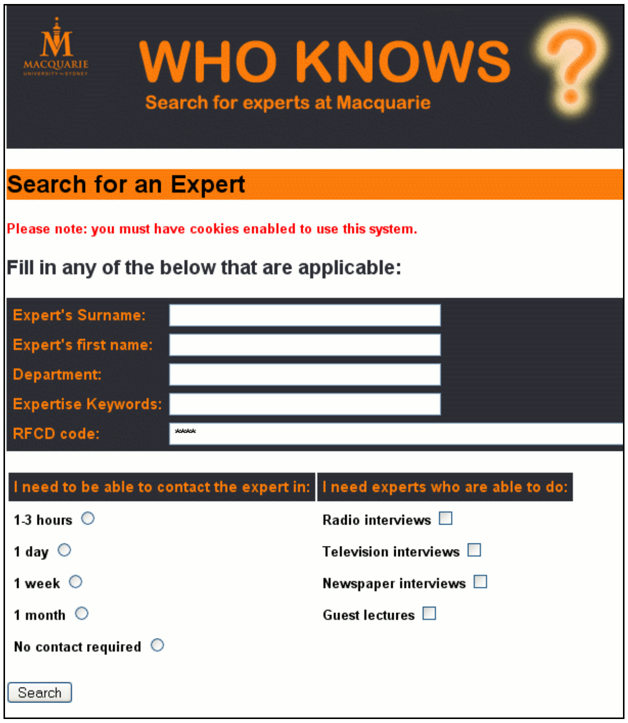
The screens which make up the user interface play a critical part of the approach by allowing the development of two profiles, or user models, of the *expert* as well as the *service requester*. The two user models will provide knowledge to the system that will be used as a filter for searching and matching. The need for customizing both sides became apparent from interviews revealing that the (type of) person being sought varied for different departments. Thus it is necessary to capture the needs and preferences of the person looking for someone (e.g., they are a news journalist and need someone to provide an expert opinion for television in layman's terms in the next hour) and also the preferences, communication skills and availability of the expert (e.g. they have given radio interviews in the past but are away for two weeks). The user models can potentially be populated from data found in other places, such as from webpages maintained by the individual or corporate pages, but realize that the feasibility of this depends on the level of webpage standardization across the organization and the degree to which content management is enforced. Thus in our solution we leave this as potential future work. Statistics relating to the responses from experts regarding their expertise and availability and the requesters satisfaction with the service provided are part of the user model. The service provided has many aspects to it, some of which, such as ability to communicate and availability, can be used to rate the expert and order future results to searches. Recommended experts who frequently receive negative feedback (or poor ratings if a rating system similar to that used by [4] is used) will be recommended less (or perhaps ranked lower in the final output) by the system, than experts who have consistently received positive feedback, for example. This feedback mechanism will act as a referral system, suggested by some of the people we spoke with as the type of system they would be interested in, rather than a yellow-pages model. They didn't want contact details of anyone professing to know a particular domain, they wanted to find someone based on the experience of someone else, who they preferably knew and trusted.

While the system will use the statistics for generating and ordering recommendations in response to a query, for privacy and ethical reasons, only the expert will be able to see his/her own statistics regarding their overall usefulness and availability as perceived by the service requesters. We see this personal feedback as potentially useful for professional development and self evaluation, similar to the way in which data from student evaluations of a teacher may be used for professional development and disclosed at the experts discretion for promotion or other reasons. As represented, this additional feed-back/validation pillar provides a triangulated approach bringing together automated machine-based knowledge discovery and manual validation.

4.1 Usability study

After the requirements for the proposed system were decided upon, it was important to test that the features of the system were actually what people wanted. All too often it is the case that a system with too many features will not be easy or enjoyable to use. Features such as an in-system contact form may be viewed as an annoyance rather than a benefit. For this reason we devised a usability test on a semi-functional version of the system.

We made a prototype expert recommender system, WHOKNOWS? that we populated with a small amount of test data. Ten mock expert profiles were put into the system as well as several screens to help searchers find and contact experts. The screens consisted of an initial search screen (Figure 7), a screen to show the results of a search, a screen for the searcher to contact the recommended experts, and a screen for the searcher to provide feedback on an expert. Screens showing the profiles of the experts were also included as well as a screen for the expert to respond to a searcher's request (Figure 6), although this screen was not included in the test.



Search for an Expert

Please note: you must have cookies enabled to use this system.

Fill in any of the below that are applicable:

Expert's Surname:
Expert's first name:
Department:
Expertise Keywords:
RFCD code: *****

I need to be able to contact the expert in:	I need experts who are able to do:
1-3 hours ○	Radio interviews □
1 day ○	Television interviews □
1 week ○	Newspaper interviews □
1 month ○	Guest lectures □
No contact required ○	

Search

Figure 7. Initial search screen in test system

WHOKNOWS? did not allow for any searcher profiles. Instead the initial search screen (Figure 7) allowed users to specify what time frame they had to contact the expert in and whether they required an expert for a radio, television, or newspaper interview, or a guest lecture.

As the system did not contain any real expert profiles, no automated searching was done by the system, rather we assumed that this stage had already been completed, and the resulting expert profiles had been stored in the system. An evaluation of automated searching and expert validation was performed on members of the Computing department at Macquarie University using publication data found in IRIS and is described in [15].

Scenario 1 Part 1.

You are a Journalist working for the Daily Star Newspaper. You are researching a story about a polar bear attacking some school children outside an Alaskan high school. You wish to find an expert on polar bears to interview for your article. You need to have the article ready in two days. Use the system to find and contact the expert.

Scenario 1 Part 2.

After contacting the expert, you receive the following reply:

Hi,
I am too busy to give any interviews at the moment. However, you may try James Paterson (james@email.com) as he will be only too happy to grant you an interview.

You contact James and he responds immediately. You are able to get an interview with him that day.

Use the system to provide feedback for the expert you initially contacted.

Figure 8. First scenario in usability test

The algorithm for ranking the recommended experts returned by a search was also implemented so it could be assessed by the participants. The algorithm contained the following steps:

1. all experts to whom the criteria entered by the searcher was not applicable were removed.
2. if the searcher entered expertise keywords, the remaining experts were ranked on how many keywords were found in their listed areas of expertise. The experts for whom no keywords were found were removed.

The remaining experts were ranked on their combined availability and searcher feedback scores.

WHOKNOWS? and the usability test were both made available online. Two scenarios were given to the participants (who responded to an emailed advertisement). Each scenario gave the participant a job occupation and a task that involved using the system to search for an expert (Figure 8). After each scenario had been completed, the participants were asked to fill in a questionnaire. The questionnaire asked the participants to rate how easy or difficult it was to complete the task, and whether the layout of the system helped or hindered them. The participants were asked to evaluate the algorithm the system uses to rank the experts, as well as their opinion on certain components of the system (such as the contact and feedback pages).

4.1.1 Participants

Thirty-eight participants responded to the emailed advertisement. However, as the usability test was online, it was not possible to make sure that the participants completed the whole test. As a result only 28 participants completed all steps which were part of the first scenario and filled out the survey, and 22 of those went on to complete the second scenario and fill out the associated survey.

	Gender	
Age Range	Female	Male
15-19		1
20-24	2	1
25-29	2	7
30-34	3	
35-39	2	2
40-44	1	1
45-49		3
50-54		
55-59		
60-64		1
65-69	1	1
Total	11	17

Table 1. Number of participants in each age range by gender

A summary of the biographical details of the 28 participants can be found in Tables 1 and 2. Participants were both male and female and a range of ages. About half the male participants and a third of the female participants were employed in a job where they needed to find experts. These included members of the Research Office and Media and Public Relations Office at Macquarie University who had participated in our interviews in section 3.2.

	Involved in finding experts		
Gender	**No**	**Yes**	**Total**
Female	7	4	11
Male	8	9	17
Total	**15**	**13**	**28**

Table 2. Number of participants who are and are not involved in finding experts for their profession by gender

4.1.2 Results and Discussion

Results were gathered for each part of the test system: the search page, results page, expert profile page, contact page and feedback page. Each of these is outlined in the subsections below. The system's ranking algorithm was also evaluated by participants, although the results are not discussed here and will be presented in a future publication.

The questionnaire given to each participant after they completed the tasks in each scenario contained statements about each screen in the system. The participants indicated their level of agreement with each of the statements on a 5-value Likert scale. The statements for each page were the following:

Search Page

S1. I found it easy to understand how to search for the expert on the search page

S2. The search options on the search page were specific enough for me to search for the expert I needed

Results Page

S1. The experts' details on the results page were sufficient for me to tell if I needed to contact them.

S2. It was clear to me how I could use the system to contact the experts on this page

S3. After reading this page I understood how to provide feedback on an expert.

Expert Profile Page

S1. The details of the expert listed on this page were sufficient for me to tell if I needed to contact them.

S2. I found the categories on this page easy to understand.

S3. It was clear to me how I could use the system to contact the expert on this page.

Contact Page

S1. It was clear to whom the email was being sent.

S2. I found it easy to understand how the text boxes should be filled in.

S3. I would use this feature in the future if available.

Feedback Page

S1. I was able to adequately express my feelings about the expert on this page.

Table 3 shows the percentage of participants that either agreed or strongly agreed with each statement for each scenario.

Search Page	Scen1	Scen2
S1	71.43%	95.45%
S2	78.57%	90.91%
Results Page		
S1	89.29%	95.45%
S2	85.71%	95.45%
S3	60.71%	86.36%
Expert Profile page		
S1	92.86%	95.45%
S2	71.43%	90.91%
S3	89.29%	95.45%
Contact page		
S1	85.71%	95.45%
S2	85.71%	90.91%
S3	89.29%	90.91%
Feedback page		
S1	60.71%	63.64%

Table 3. Percentage of participants who agreed or strongly agreed with statements about each screen in the system after completing tasks in the first and second scenarios

From Table 3 we can see that the percentage of people who agreed or strongly agreed with the statements after completing the task in scenario 2, is higher in every case than the percentage who agreed or strongly agreed after completing the task in scenario 1.

The percentage increase can be explained partly by the fact that the participants would have a better grasp of how the system works after completing the first scenario and starting the second; and partly by the fact that six of the participants who completed the first scenario did not complete the second. Table 4 shows the percentage of the 22 participants who completed both scenarios who either agreed or strongly agreed with each statement for each scenario

After removing the 6 participants who didn't continue to the second scenario, we can see in Table 4 that the percentage differences are smaller in general than in Table 3. Percentage increases are recorded for both statements about the Search page; statement 3 about the Results page; and statement 2 about the Expert Profile page. This increase can most likely be attributed to the participants gaining experience in using the system after they completed the second scenario

Search Page	Scen1	Scen2
S1	81.82%	95.45%
S2	86.36%	90.91%
Results Page		
S1	95.45%	95.45%
S2	95.45%	95.45%
S3	68.18%	86.36%
Expert Profile Page		
S1	100.00%	95.45%
S2	81.82%	90.91%
S3	100.00%	95.45%
Contact Page		
S1	95.45%	95.45%
S2	90.91%	90.91%
S3	90.91%	90.91%
Feedback Page		
S1	68.18%	63.64%

Table 4. Percentage of participants who agreed or strongly agreed with statements about each screen in the system after completing tasks in the first and second scenarios with participants who did not complete both tasks removed.

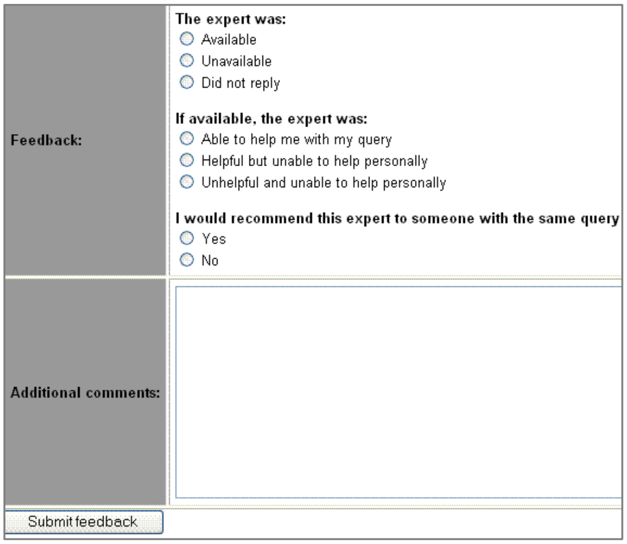
A few participants found the layout of the search page confusing initially, and some said that they would have preferred fewer options and an "advanced search" option instead.

A relatively low percentage of people (68.18%) agreed or strongly agreed with the third statement about the Results Page (*after reading this page I understood how to provide feedback on an expert*) after completing the first task. Many participants thought the instructions on how to submit feedback were not very clear initially. To rectify this, there should be a separate button for each expert that, when clicked, would take the searcher immediately to the feedback page for that expert. In reality, however, a searcher is probably not likely to provide feedback on an expert immediately, but rather after some time has elapsed and they have been sent a reminder email by the system that includes a link to the feedback page for the expert they contacted.

One participant commented that they were not able to discern how helpful an expert was going to be by viewing their profile. This is a difficult problem to fix, as the feedback information the system uses to rank an expert is not displayed for ethical and practical reasons. Many experts would not be happy with their details in a system that displays to the public what other people think of them. If an expert saw that they had an average feedback score of 20%, for instance, they may become upset and ask for their profile to be removed from the system. While an expert should be allowed to know what their feedback score is, showing this information to all users of the system would not be appropriate. Showing an expert's availability information to the public, however, may be acceptable, as this is not based on people's opinions, but on facts.

It would be beneficial to both the searcher and the expert to have the expert's availability information displayed, as the searcher will know that they might not have much luck if they try to contact the expert, and the expert will not have to be constantly rejecting requests for help from searchers.

The percentage of people who agreed or strongly agreed with the statements about the Contact page remained the same for both tasks. The most promising result was the high percentage of participants who indicated for both scenarios that they would use the feature in the future if available (statement 3). None of the participants disagreed with this statement, although one participant commented that they could imagine copying the email address and sending their own email to the expert.



Feedback:	The expert was: ○ Available ○ Unavailable ○ Did not reply If available, the expert was: ○ Able to help me with my query ○ Helpful but unable to help personally ○ Unhelpful and unable to help personally I would recommend this expert to someone with the same query ○ Yes ○ No
Additional comments:	

Figure 9. Feedback form

The statements about the feedback page generated the lowest percentage of agreement (68.18% after the first task and 63.64% after the second task). Some participants thought the feedback options available on this page (Figure 9) were too rigid, especially for scenario 1, when the recommended expert actually recommended another expert, but wasn't of any help otherwise. The section of the feedback form that requires a Yes/No answer (*I would recommend this expert to someone with the same query*) would be especially hard to fill out in a situation such as this. Another participant commented that the additional comments section was the only place where an expert's performance could be evaluated (with the other sections evaluating the expert's immediate response and availability). The feedback page was structured in this way to avoid making people fill out too many sections, as they would be unlikely to provide feedback regularly if this was the case.

Adding another section to indicate how satisfactory an expert's performance was could be a step towards solving this problem. It could allow the user to give a Yes/No response to the statement "I was satisfied with the expert's performance" or have them rate their satisfaction with the expert's performance on a scale of 1-5 with 5 being very satisfied and 1 being very dissatisfied. There are some issues with this method, however. A person's satisfaction with another person's performance can be very subjective. One person may think an expert performed excellently, while another may think they performed poorly, even if they gave the same performance in both cases. If free text was used to evaluate an expert's performance, the searcher could choose the comments to be sent to the expert so they can see exactly what the searcher was dissatisfied with. A Yes/No response, or a rating out of 5 would not give the expert a good idea of exactly what the searcher thought, and would therefore not be able to improve.

A second option would be to show each expert their feedback scores and comments on a private part of their profile. This may encourage them to improve their performance if the searcher was not satisfied. This would require comments to be heavily moderated, however, to ensure that searchers are not allowed to submit abusive feedback.

5. Conclusion

Rating systems, such as we propose and that used by [4], raise several ethical issues. For instance many people may object to the concept of rating another person and may refuse to participate. On the other end of the scale, some users may give recommended experts an unnecessarily bad rating simply because they don't like them on a personal level. In addition, the possibility that a person's personal or teaching skills could be criticized would be a sensitive issue for many and may result in a large number of experts refusing to be registered in the system. In our approach we aim to try different methods of user feedback as well as limiting the visibility of an expert's feedback results and preserving the anonymity of the users who provide the feedback in an attempt to avoid the ethical issues. In addition, we will also consider the use of more personalised feedback (e.g. a reporter wishing to interview an expert will be rating them on different criteria than someone wishing to work with the expert on a project).

As a key part of our approach is the combination of self-reporting/referral and automated searching through available data. Some data can only be obtained via self-reporting (e.g. indicating if you are available to do media interviews or guest lectures). However, information about which units one teaches, expertise areas, grants, etc. can be gained from personal websites and internal databases. An outstanding issue would be how to reconcile differences between these sources and between the outputs of automated searching and self-reporting.

A usability study was performed on the test expert recommender system we developed – WHOKNOWS? The participants included both males and females across a range of ages, a number of whom are concerned with finding experts in their professions. The participants completed two tasks using the test system and filled in questionnaires about the system’s various features. A large majority of the participants responded favourably to the system and provided valuable feedback that resulted in a re-evaluation/design of the system’s ranking algorithm, search options and feedback form.

Recommender systems greatly speed up and simplify the searching process, whether the item being searched for is a book, film, or another person. This project is interested in a recommender system which maintains user profiles to match experts with service requesters. Validation of the user profiles and the system recommendations using a combination of automated and human-based techniques seeks to combine and reinforce these two main approaches which individually have numerous weaknesses. Closing the loop between the seeker and the sought is aimed at providing both parties with confidence and motivation to use the system. We anticipate that our findings will be of use to other recommender systems and search engines, such as Google, in general. Finally, to provide a generalized framework for expertise location, we will consider what modifications are necessary to allow other knowledge-intensive organizations to use the framework and toolkit.

6. References

[1] D. Richards, M. Taylor, and P. Busch, “Expertise Recommendation: A two-way communication channel”, Proc. 4th International Conference on Autonomic and Autonomous Systems (ICAS’08), March 16-21, 2008, Gosier, Guadeloupe, pp. 35-40.

[2] D. Yimam-Seid, and A. Kobsa, “Expert Finding Systems for Organizations: Problems and Domain Analysis and the DEMOIR approach” Jrnl Org.l Comp. & Electronic Commerce vol. 13, 2003, pp. 1-24.

[3] Y. Sim, R. Crowder and G. Wills, “Expert Finding by Capturing Organizational Knowledge from Legacy Documents” Proc. IEEE ICCCE '06, Kuala Lumpur, Malaysia, 2006.

[4] E. Aïmeur, F. Onana, F. S. Mani and A. Saleman, “HELP: A Recommender System to Locate Expertise in Organizational Memories” Proc. AICCSA 2007. pp. 866-874.

[5] L. Streeter, and Lochbaum, K. “An Expert/Expert Locating System Based on Automatic Representation of Semantic Structure” Proc. 4th IEEE Conf. on AI Appl. Comp. Soc. of IEEE, San Diego CA, 1998, pp. 345-349.

[6] I. Becerra-Fernandez, “The Role of Artificial Intelligent Technologies in the Implementation of People-Finder Knowledge Management Systems” Knowledge-Based Systems, vol. 13, 2000, pp. 315-320.

[7] H. Kautz, B. Selman, and M. Shah, “Referral Web: Combining Social Networks and Collaborative Filtering” Communications of ACM, vol. 40(3), 1997, pp. 63-65.

[8] J. Lave, and E. Wenger, Situated Learning: Legitimate Peripheral Participation, Cambridge University Press, Cambridge, U.K, 1991.

[9] L. Foner, “Yenta: A Multi-Agent Referral-Based Matchmaking System” Proc. 1st Int.l Conf. on Autonomous Agents, Marina del Rey California, 2002, pp. 301-307.

[10] A. Pikrakis, T. Bitsikas, S. Sfakianakis, M. Hatzopoulos, D. DeRoure, S. Reich, G. Hill, and M. Stairmand, “MEMOIR – Software Agents for Finding Similar Users by Trails” Proc. 3rd Intl. Conf. on Practical Application of Intelligent Agents and Multi-agents, London, UK, 1998, pp. 453-466.

[11] D. McDonald, and M. Ackerman, “Expertise Recommender: A Flexible Recommendation System and Architecture” Proc. ACM2000 CSCW Conf., Philadelphia PA, 2000, pp. 231-240.

[12] A. Vivacqua, “Agents for Expertise Location” in Proc. AAAI Spring Symp. on Intelligent Agents in Cyberspace, Stanford, CA, 1998, pp. 9-13.

[13] R. Crowder, G. Hughes and W. Hall, “An agent based approach to finding expertise” in Proc. 4th Int.l Conf. on Practical Aspects of Knowledge Mgt Heidelberg Germany , 2002, pp. 179-188.

[14] S. E. Middleton, D. C. D. De Roure, and N. R. Shadbolt, “Capturing knowledge of user preferences: ontologies in recommender systems”. Proc. International Conference on Knowledge Capture (K-CAP’2001), ACM Press, New York, NY, USA, 2001, pp. 100-107.

[15] M. Taylor, and D. Richards, Discovering Areas of Expertise from Publication Data, In B-H. Kang and D. Richards, Proceedings of Pacific Knowledge Acquisition Workshop in conjunction with PRICAI'08, December 16-17, Hanoi, Vietnam, 2008, pp. 173-186.

Assurance-driven design in Problem Oriented Engineering*

Jon G. Hall Lucia Rapanotti
Department of Computing
The Open University, UK
{J.G.Hall,L.Rapanotti}@open.ac.uk

Abstract

The design of assurance cases is hampered by the posit-and-prove approach to software and systems engineering; it has been observed that, traditionally, a product is produced and then evidence from the development is looked for to build an assurance case. Although post-hoc assured development is possible, it often results in errors being uncovered late—leading to costly redevelopment—or to systems being over-engineered—which also escalates cost. As a consequence, there has been a recent move towards the proactive design of the assurance case. Assurance-driven design sees assurance as a driving force in design. Assurance-driven design is suggestive of how the design process should be shaped for assurance. It is not, however, a prescriptive method; rather it allows an organisation to assess their assurance needs according to their developmental needs, including their attitude to risk, and to adapt their processes accordingly.

We have situated the work within Problem Oriented Engineering, a design framework inspired by Gentzen-style systems, with its root in requirement and software engineering. In the paper we present the main elements of the approach and report on its application in real-world projects.

Keywords: Dependability, Software Engineering, Assurance Case, Problem Oriented Engineering, Engineering Design

1 Introduction

By engineering design (shortly, design), we refer to the creative, iterative and often open-ended endevour of conceiving and developing products, systems and processes (adapted from [2]).

Engineering design by necessity includes the identification and clarification of requirements, the understanding and structuring of the context into which the engineered system will be deployed, the detailing of a design for a solution that can ensure satisfaction of the requirements in context, and the construction of arguments to assure the validating stake-holders that the solution will provide the functionality and qualities that are needed. The last of these is the concern of this paper.

Typically, for software at least, even though evidence is gathered during development the collation, documentation and quality injection of the assurance argument follows construction; perhaps this is because software development is currently sufficiently difficult without having to serve the needs of two masters: code *and* assurance. If software and assurance argument could be developed together, then developmental risk could be managed better–development errors that weaken an assurance argument could be found earlier in the process—as could developmental cost—by removing the compensating tendency to over-engineer.

Assurance-driven design (ADD), introduced in [1], does not make development any simpler; rather, it makes the building of an assurance argument a driver for development. Accepting this, however, ADD can guide the developer: by providing a more specific focus on those parts of a system that *require* assurance; by providing early feedback on design decisions; by capturing coverage of the design space; and, last but not least, by delivering an assurance argument alongside the product.

Our work on assurance-driven design is situated within Problem Oriented Engineering (POE), our framework for engineering design (instantiated for software in [3, 4]). The techniques we propose have no particular dependence on a software development context; indeed, our main example combines software and educational materials design and it is the assurance of their combined qualities that will drive our development.

The paper is structured as follows. Section 2 provides the briefest introduction to POE. In Section 3 we develop assurance-driven design, and in Section 4 illustrate its use through its application to a real-world problem. Section 5

[*] An expanded version of [1]

relates our work to that of others, and Section 6 reflects on what has been achieved and concludes the paper.

2 Problem Oriented Engineering

Problem Oriented Engineering is a framework for engineering design, similar in intent to Gentzen's Natural Deduction [5], presented as a sequent calculus. As such, POE supports rather than guides its user as to the particular sequence of design steps that will be used; the user choosing the sequence of steps that they deem most appropriate to the context of application. The basis of POE is the *problem* for representing *design problems* requiring designed solutions. *Problem transformations* transform problems into others in ways that preserve solutions (in a sense that will become clear). When we have managed to transform a problem to axioms[1] we have solved the problem, and we will have a designed solution for our efforts. A comprehensive presentation of POE is beyond he scope of this paper and can be found in [3, 4].

2.1 Problem

A problem has three descriptive elements: that of an existing real-world problem context, W; that of a requirement, R; and that of a solution, S. We write the problem with elements W, S and R as $\mathsf{W}, \mathsf{S} \vdash \mathsf{R}$. What is known of a problem element is captured in its description; descriptions can be written in any appropriate language: examples include natural language, Alloy ([6]), and machine language. Solving a problem is finding S that satisfies R in the context of W.

Figure 1 gives an example of engineering design problem (shortly problem), described in a Problem-Frame-like notation ([7]). The problem (from a real world case study [8, 9]) is that of defining a controller to release decoy flare from a military aircraft: essentially decoy flares provide defence against incoming missile attack. The context includes a Pilot, a Defence system and some other existing hardware, represented in the figure as named undecorated rectangles. The solution to be designed is Decoy Controller, represented as a named decorated rectangle. The arc annotations are shared phenomena: for instance, the Pilot can send an ok command to the Decoy Controller. The solution needs to satisfy the Safe decoy control requirement, represented as a named dotted ellipse, for the safe release of decoys. Formally, in POE, this problem is represented as:

$$\begin{aligned}&\mathsf{Defence\ System}^{\mathsf{con}}, \mathsf{Dispenser\ Unit}^{\mathsf{out}}_{\mathsf{fire,sel}},\\ &\quad \mathsf{Aircraft\ Status\ System}^{\mathsf{air}},\\ &\qquad \mathsf{Pilot}^{\mathsf{ok}}, \mathsf{Decoy\ Controller}^{\mathsf{fire,sel}}_{\mathsf{con,out,air,ok}} \vdash \mathsf{SDC}^{\mathsf{fire,sel}}_{\mathsf{con,out,air,ok}}\end{aligned}$$

but we use both notations interchangeably.

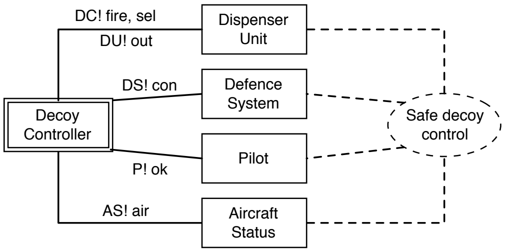


Figure 1. The Decoy Controller Problem

2.2 Problem transformations and justification obligations

Problem transformations capture discrete steps in the problem solving process. Many classes of transformation are recognised in POE, reflecting a variety of engineering practices reported in the literature or observed elsewhere. Problem transformations relate a problem and a justification to (a set of) problems. Problem transformations conform to the following general pattern (whose notation is based on that of [5]). Suppose we have *conclusion* problem $\mathsf{P} : \mathsf{W}, \mathsf{S} \vdash \mathsf{R}$, *premise* problems $\mathsf{P_i} : \mathsf{W_i}, \mathsf{S_i} \vdash \mathsf{R_i}$, $\mathsf{i} = 1, ..., \mathsf{n}$, $(\mathsf{n} \geq 0)$ and *justification* J, then we will write:

$$\frac{\mathsf{P_1} : \mathsf{W_1}, \mathsf{S_1} \vdash \mathsf{R_1} \qquad \dots \qquad \mathsf{P_n} : \mathsf{W_n}, \mathsf{S_n} \vdash \mathsf{R_n}}{\mathsf{P} : \mathsf{W}, \mathsf{S} \vdash \mathsf{R}} \begin{matrix}[\textsc{Name}]\\ \langle\langle \mathrm{J} \rangle\rangle\end{matrix}$$

to mean that, derived from an application of the Name problem transformation schema (discussed below):

> S is a solution of $\mathsf{W}, \mathsf{S} \vdash \mathsf{R}$ with *adequacy argument* $(\mathsf{AA_1} \wedge ... \wedge \mathsf{AA_n}) \wedge \mathsf{J}$ whenever $\mathsf{S_1}, ..., \mathsf{S_n}$ are solutions of $\mathsf{W_1}, \mathsf{S_1} \vdash \mathsf{R_1}, ..., \mathsf{W_n}, \mathsf{S_n} \vdash \mathsf{R_n}$, with adequacy arguments $\mathsf{AA_1}, ..., \mathsf{AA_n}$, respectively.

Engineering design under POE proceeds in a step-wise manner with the application of problem transformation schemata, examples of which appear below: the initial problem forms the root of a *development tree* with transformations applied to extend the tree upwards towards its leaves. A problem is solved for a stake-holder S if the development tree is complete, and the adequacy argument constructed for that tree convinces S that the solution is adequate. For technical reasons[2], we write

$$\overline{\mathsf{P}}$$

to indicate that problem $\mathsf{P} = \mathsf{W}, \mathsf{S} \vdash \mathsf{R}$ is solved. As P will be fully detailed in determining the solution — to the satisfaction of stake-holders — the indication that P is solved is without justification. For the technical details, see [4].

A partial development tree is shown in Figure 2.

[1] An *axiomatic problem* is a problem whose adequate, i.e., fit-for-purpose, solution is already known.

[2] Simply, that we may indicate an axiom in a Gentzen system thus.

$$\dfrac{\dfrac{\overline{\mathsf{P}_3 : \mathsf{W}_3, \mathsf{S}_3 \vdash \mathsf{R}_3} \qquad \mathsf{P}_4 : \mathsf{W}_4, \mathsf{S}_4 \vdash \mathsf{R}_4}{\mathsf{P}_2 : \mathsf{W}_2, \mathsf{S}_2 \vdash \mathsf{R}_2}\,{\scriptstyle [\mathsf{N}_2]\,\langle\langle \mathsf{J}_2 \rangle\rangle}}{\mathsf{P}_1 : \mathsf{W}_1, \mathsf{S}_2 \vdash \mathsf{R}_1}\,{\scriptstyle [\mathsf{N}_1]\,\langle\langle \mathsf{J}_1 \rangle\rangle}$$

Figure 2. A POE partial development tree

The figure contains four nodes, one for each of the problems P_1, P_2, P_3 and P_4. The problem transformation that gave the problem solver P_1 is justified by J_1, whereas the branching to problems P_2 and P_3 is justified by J_2. From the tree, we see that P_3 is solved. P_4 remains unsolved, so that the adequacy argument for the tree is incomplete; from the definition above, the incomplete adequacy argument is:

$$\mathsf{J}_2 \wedge \mathsf{J}_1$$

2.2.1 "Have we done enough?"

At any point in a development we can ask if we have done enough, i.e., if we were to declare our development complete would we be able to satisfy the validating stake-holders? This question is most obviously asked of the complete development, in which case an affirmative answer convinces all stake-holders that we have an adequate solution to the whole problem.

As previously mentioned, a completed development in POE is represented as a complete development tree, i.e., a tree in which no problems exist to be solved. The development is successful if the adequacy argument, AA, satisfies the stake-holders of the adequacy of the solution. For any stake-holder S, then, we have done enough if

$$\mathsf{AA} \text{ convinces } \mathsf{S}.$$

Consider again the form of the adequacy argument given a partial tree, such as that in Figure 2. Suppose that S is a stake-holder for problem P_1. Should P_1 be solvable, we would wish to find justification J_3 and solved problem P_5, say, such that

$$\mathsf{J}_3 \wedge \mathsf{J}_2 \wedge \mathsf{J}_1 \text{ convinces } \mathsf{S} \qquad \text{and} \qquad \dfrac{\overline{\mathsf{P}_5}}{\mathsf{P}_4}\,{\scriptstyle [\mathsf{N}_3]\,\langle\langle \mathsf{J}_3 \rangle\rangle}$$

If we were free to choose P_5 without any reference to the requirements of the argument that establishes it as fit-for-purpose (that formed when J_3 is added to the adequacy argument) it would be unlikely to result in something that could be justified. Of the techniques mentioned in the introduction to this paper, the 'posit' of 'posit and prove' is moving towards this 'free' choice; moreover, over-engineering a solution simply allows the engineer a freer choice.

As we begin to balance the choice of P_5 and J_3, we move towards the position of assurance-driven design, in which the requirements for justification motivate the design. The techniques we introduce in this paper allow us to structure the development so that this balance can occur. Primary amongst them is the construction of projections from the overall development tree into, what might be called, 'stake-holder spaces' in which validation takes place.

2.2.2 A formal backwater: the weakest pre-justification

Although it is — currently — only of theoretical interest, by inspection, there is a best such justification that, given an incomplete development tree, completes the adequacy argument so as to just satisfy the stake-holder, S. By analogy to other formal systems, we term this the *weakest pre-justification*, $\mathsf{J}_{\mathsf{wpj}}$, such that, if IAA is the current (incomplete) adequacy argument for that tree, then for any K

$$(\mathsf{K} \wedge \mathsf{IAA} \text{ convinces } \mathsf{S}) \Rightarrow (\mathsf{K} \Rightarrow \mathsf{J}_{\mathsf{wpj}})$$

3 Assurance-driven design

A metaphor for engineering design under POE is that one grows a forest of trees. Each tree in the forest grows from a root problem through problem transformations that generate problems like branches; with happy resonance, the tree's *stake*-holders guide the growth of the tree. Some trees, those that have root problems that are validatably solvable for its stake-holders will grow until they end with solved problem leaves.

There are many reasons why the forest has many trees: described elsewhere [10], but only of note in this paper, is the preservation of a record of unsuccessful design steps, i.e., design steps that are not validatable for the current stake-holders, which cause a development to backtrack to a point where a different approach can be taken. The backtracked sub-trees are kept as record of unsuccessful development strategies[3].

For this paper, we note simply that development trees grow through the developer's careful choice of effective design steps. To produce an effective design step, the developer must consider both the problem(s) that the step will produce towards solution *and* what is the justification obligation that will satisfy the *validating stake-holders*. With the discharged justification obligations forming the basis of the adequacy argument, the result of a sequence of effective design steps is a solution *together with its assurance*

[3] Backtracked trees are not 'deadwood'; rather they stand as proof of design space exploration, with their structure being reusable for, for instance, other stake-holders' problems. Unsolved problems that remain in backtracked trees do not affect the completed status of a development.

argument[4]. We have observed the interplay of design steps and their justification under POE (for instance, [11]), and have developed a simple, composable process pattern—the POE Process Pattern—that guides their effective interleaving. The structuring of the problem solving activity through the POE process pattern is the basis of assurance-driven design.

We note that a problem transformation schema is applied to a conclusion problem, and that the development tree is extended up by the application. There is no necessity for any premise problem to be determined before the justification is added. Indeed, one could see the problem solver saying "It is fashionable to have a fan oven in stainless steel", and then searching for an oven that fits the bill[5].

It is determining the needs for the justification, rather than for the premise problem(s), that motivates us to introduce assurance-driven design: assurance-driven design determines the justification first, and then looks for the corresponding premise problem.

3.1 The POE process pattern

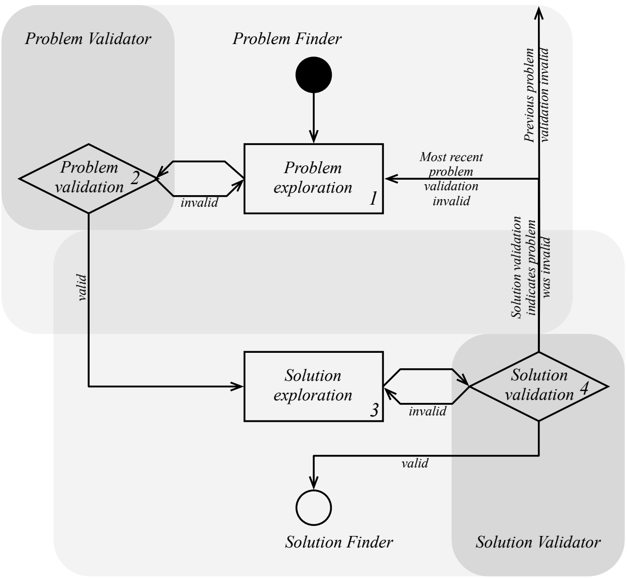


Figure 3. The POE Process Pattern for assurance-driven design

The POE process pattern shown in Figure 3 is described in a variant of the UML activity diagram notation [12]: rectangles are resource consuming activities; diamonds indicate choice points; the flow of information is indicated by arrows; the scope of the various roles is indicated by shading, overlapping shading indicating potential need for communication between roles. Referring to the numbers in the figure: first explore the problem better to understand it (*1*), checking that understanding through problem validation (*2*), iterating problem exploration as necessary; then explore the solution better to understand its design (*3*), checking that understanding through solution validation (*4*), iterating solution exploration as necessary.

The role of a *problem finder* during problem exploration is to explore their understanding of the problem (or part thereof), perhaps with the help of others. The goal of problem exploration is to produce descriptions of the problem that will satisfy the problem-validators(s) at problem validation. Similarly, the role of *solution finder* during solution exploration is to explore their understanding the solution (or part thereof) to the problem, again perhaps with the help of others. The goal of solution exploration is to produce descriptions of the solution that will satisfy the solution validator(s) at solution validation.

The role of a *problem validator* is to validate a candidate problem description. There are many familiar examples of problem validator. These include, but are not limited to:

- the customer or client — those that pay for a product or service;
- the regulator — those whose remit is the certification of safety of a safety of a safety-critical product, for instance;
- the business analyst — whose role is to determine whether the problem lies within the development organisation's business expertise envelope;
- the end-user — those who will use the product or service when commissioned.

It is a problem validator's role to answer the question "Is this (partial) problem description valid?" Depending on a problem validator's answer, the Problem Finder will need to re-explore the problem (when the answer is "No!"), or task the Solution Finder to find a (partial) solution (when the answer is "Yes!").

The role of the *solution validator(s)* is to validate a (candidate or partial) solution description, such as a candidate architecture (a partial solution) or choice of component (something of complete functionality). Although present in every commercial development, the roles of solution validator may be less familiar to the reader. They include, but are not limited to:

- a technical analyst — whose role is to determine whether a proffered solution is within the development organisation's technology expertise envelope;

[4] If there are no validating stake-holders for a development, the justification obligations can be ignored.

[5] Of course, we could have written such a statement as part of the requirement, but that would have been the stake-holder's statement, not the problem solver's.

- an oracle — who determines, for instance, which of a number of features should be included in the next release;

- a unit, module, or system tester; a project manager— who needs to timebox particular activities.

It is the solution validator's role to answer the question "Is this (candidate or partial) solution description valid?" Depending on their response, the problem solver may need to re-explore the solution (when the answer is "No!"), move back to exploring this or a previous problem (when the answer is "No, but it throws new light on the problem!"), or moving on to the next problem stage (when the answer is "Yes!").

The potential for looping in the POE process pattern concerns unsuccessful attempts to validate, and is indicated by arrows labelled *invalid* in the figure. Those leading back to exploration activities, of which there are two, continue their respective exploration activities in the obvious way. The other two invalid arrows lead from a failed solution validation to restart a problem exploration when the indication is that it was wrong. Examples of this latter form of failure are well known in the literature. For instance, Don Firesmith, in an upcoming book [13], talks about the need for architecture *re*-engineering in the light of inadequately specified quality requirements [part of an earlier problem exploration]:

> [...] it is often not until late in the project that the stakeholders recognize that the achieved levels of quality are inadequate. By then [...] the architecture has been essentially completed [solution exploration], and the system design and implementation has been based on the inadequate architecture.

In this way, recognising late that inadequately specified quality requirements (as discovered through problem exploration and validated at problem validation) have not been met can be very difficult and expensive to fix; leading to revisiting a long past problem, that of re-establishing the architecturally significant quality requirements[6].

Although we do not consider developmental risk explicitly in this paper, we note that feedback within the process has an impact on resources: an unsuccessful validation indicates that some previous exploration was invalid, to a greater or lesser extent. Moreover, some proportion of the development resource that will have expended during and subsequent to that exploration — the impact of the failed validation — will have been lost[7].

After successful problem validation, handover between the problem and solution finders occurs. In problem and solution finder are the same person, this raises no issues. Otherwise, it is possible to consider the solution finder as a problem validator, so that they receive a description of the problem that they have validated as the basis of their solution exploration. Symmetry dictates that the problem finder should have a role in solution finding too.

3.1.1 Building potent design processes

Although the POE process pattern provides a structure for problem solving, in its raw form, a problem will only be solved (i.e., the end state in Figure 3 is reached) when, after iteration, a validated problem is provided with a validated solution. This 'bang-bang' approach is suitable for simple problems, but is unlikely to form the basis of any realistically complex problem encountered in software engineering.

To add the necessary complexity, the POE process pattern combines with itself in three basic ways; in combination, it is again a process that can be combined. The three ways it can be combined are in sequence, in parallel and in a fractal-like manner, as suggested in Figure 4, and as described in the sequel.

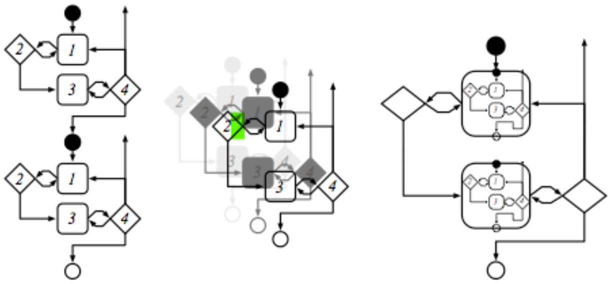


Figure 4. (a) Sequential, (b) Parallel and (c) Fractal-like combination

Sequential Design By identifying the end of one complete problem solving cycle with the start of another (see Figure 4(a)), we move a partially solved problem to the next phase: using the validated solution to explore the problem further. In [4], we show how a partial solution in the form of an architecture can lead to more detailed problem exploration: in that paper, we use the Model-View-Controller architecture to structure the solution of a problem, simplifying the problem to one of defining first the Model, then the View and finally the Controller.

In sequence, the POE process pattern models (more or less traditional) design processes in which architectures are

[6] Firesmith cites Boeing's selection of the Pratt and Whitneys PW2037 Engine for the Boeing 757 [14] as an instance of this problem.

[7] Work on risk management in POE is in preparation at the time of writing.

used as structure in the solution space according to architecturally significant requirement and qualities, and according to developmental requirements.

Parallel Design By identifying many instances of the POE process pattern through the start state, many problem solvers can solve problems in parallel. Architectures that admit such concurrent problem solving, and that might be discovered in a sequential prelude to such a process, are evident in many areas. One of timely relevance, given their current popularity, is open source projects, such as the GNU Classpath project whose goal is to provide

> 'a 100% free, clean room implementation of the standard class libraries for compilers and runtime environments for the java programming language.'

Concurrent development may place demands on the resources shared throughout the concurrent design. For instance, during problem and solution validation should access to the various stake-holders be co-ordinated, or should individual problem and solution finders be allowed access to them as and when necessary?

Communications between those involved in parallel development is an issue on the GNU Classpath project, and it is not surprising that explicit guidance exists to i) partition work through a task list and a mailing list, ii) contact the central maintainer of the project when the developer wishes to make certain non-trivial additions to the project, iii) global announcements whenever important bugs are fixed or when 'nice new functionality' is introduced.

Fractal-like Design Fractal structures are self-similar in the sense that the whole structure resembles the parts it is made of [15]. Another way to look at it is that the whole is generated from simple building blocks, with complexity emerging through recursion of the simple generators. By analogy, problem solving under the POE process pattern is structurally simple and admits recursive application in that problem solving activity can occur in the Problem Exploration and Solution Exploration parts of the POE process pattern. In the next section, we show how this leads to our notion of assurance-driven design.

3.1.2 The 'fractal' nature of validation

Given that problem and solution exploration can both be instances of the POE process pattern, let us consider the problems and solutions they work with.

As Problem Exploration leads to Problem Validation, it is 'complete' when we have delivered a problem description that satisfies the problem validator. That is, Problem Exploration is complete when we have found a Problem Description that solves the following *problem validation problem*, illustrated in Figure 5(a):

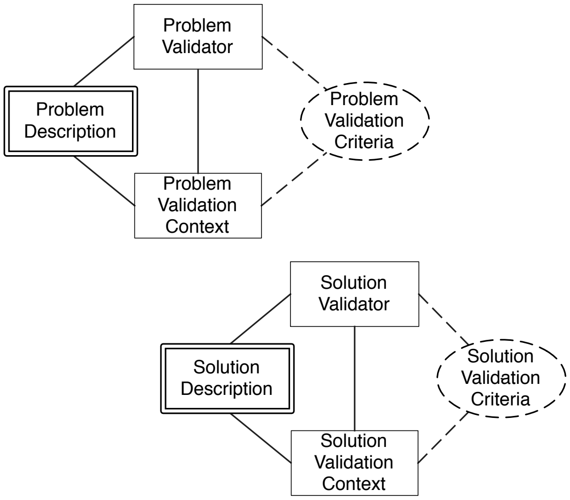

Problem Validator
Problem Description
Problem Validation Criteria
Problem Validation Context
Solution Validator
Solution Description
Solution Validation Criteria
Solution Validation Context

Figure 5. (a) Problem exploration as a problem validation problem, and (b) Solution exploration as a solution validation problem.

$$\text{Problem Validation Context, Problem Validator,}\\ \text{Problem Description} \vdash \text{Problem Validation Requirements}$$

The Problem Validation Context (PVC) is a description of the context in which the validation of the problem will be undertaken, and will need to be found as part of the fractal problem exploration phase of the outer problem exploration, as will the Problem Validation Requirements (PVR), i.e., the requirements that will need to be met for the problem to be validated. Note that the Problem Validator (PV) is an explicit domain in the context.

Symmetrically, solution exploration can be seen as complete when we have found a Solution Description that, when considered in the Solution Validation Context (SVC), satisfies the Solution Validation Requirements (SVR) of the Solution Validator (SV). As a POE problem, this is the *solution validation problem*, illustrated in Figure 5(b):

$$\text{Solution Validation Context, Solution Validator,}\\ \text{Solution Description} \vdash \text{Solution Validation Requirements}$$

Although the fractal-like nature makes an easy clarity somewhat difficult, the view we have just presented fits well with practice. Indeed, discussions that lead to an agreed (i.e., validated) collection of use-cases [16] can be seen as a technique for producing a problem description that satisfies the problem validation problem. Moreover, discussions that lead to an agreed collection of acceptance tests can be seen as a solution description that satisfies the solution validation problem. Requirements engineering, consisting of elicitation, analysis, specification can be seen as a technique for partial problem exploration; pattern oriented analysis and design is a technique for partial solution exploration.

In terms of Section 2.2, each validation is a projection of a whole development tree's adequacy argument into the stake-holder space determined by the validation context and validation requirements and, for a properly engineered solution, considering each of the adequacy problems is important.

4 Assurance-driven design in practice

The companion paper, [1], presented the assurance-driven design of a safety-critical subsystem of an aircraft. In this paper, we present a very different problem, that of the assurance-driven design of a research programme for The Open University. Whereas the aircraft example involved just a single stake-holder — the regulator for the system — this paper's example involved over 50 stake-holders as problem and solution validators. The project manager for the programme is the second author. For more information about that project, and a discussion of how POE was adopted in practice, please refer to [17, 18].

4.1 Notation

Because of the needs of the problem, we have augmented the traditional Gentzen-style notation to support better the separation of the problem and solution explorations, and to link validation problems to the justification of which they form a part. Figure 6 illustrates the differences. In the figure, we see the traditional transformations involving the problems labelled 'design problem' that will be familiar from Section 2.2. The triangular structures that extend the horizontal bar indicates the collection of validation problems associated with the step: by convention, when written on the right they are problem validation problem, when on the left they are solution validation problem[8].

4.2 Example

The Computing Department at The Open University is in the process of developing a new part-time MPhil programme to be delivered at a distance, supported by a blend of synchronous and asynchronous internet and web technologies — the *eMPhil*. The eMPhil is innovative in many ways in its adoption and use of emergent technology, like Second Life and Moodle, to support the core processes of the programme (the interested reader is directed to [19] for details).

The eMPhil project team was faced with a complex socio-technical problem, that of the adoption and development of appropriate software systems and the definition of new processes and practices, of the design and delivery of induction and training activities for staff and students, and the institution of a framework for quality assurance, monitoring and continuous process improvement. The project also found itself with many stake-holder groups, those who would play problem validators, such as the Head of the Department of Computing, and solution validators, including Head of the Research Degrees Committee and Pro-Vice Chancellor for Research and Enterprise. The difficulties of managing the design and validation of the programme partially motivated the development of and application of the techniques described in this paper.

4.2.1 The problem

The eMPhil was required to meet a number of objectives:

- for the Head of the Department of Computing: to enhance and develop the department's provision to its graduate community; to increase the overall amount of research supervision that takes place within the department;
- for at-a-distance students: to make available technology for their support; to provide as a forum for that student community; to allow those unable to commit time for a PhD a research degree to study for;
- for academics wishing to promote research in their area: to create cohorts of research students on specific research themes and projects;
- for the Head of the Research Degrees Committee and Pro-Vice Chancellor Research and Enterprise: to support the development of research skills; to comply with university policy on research student induction and training; to comply with national standards [20].

The eMPhil core project team — the problem and solution finders — was composed of four academics, with the second author as project leader. The POE process pattern was used as described below to shape the project, with the techniques described earlier in the paper used to drive and manage its development and risks, as well as to identify the eMPhil project's needs for resource and communication.

4.2.2 The process

Figure 6 illustrates the early design steps taken by the development team towards a solution to the problem. The first transformation (bottom of the figure) achieves a first characterization of the problem context and requirement (from

[8] Because of the separation of problem and solution validations, never will problem and solution validations need to appear in the same step.

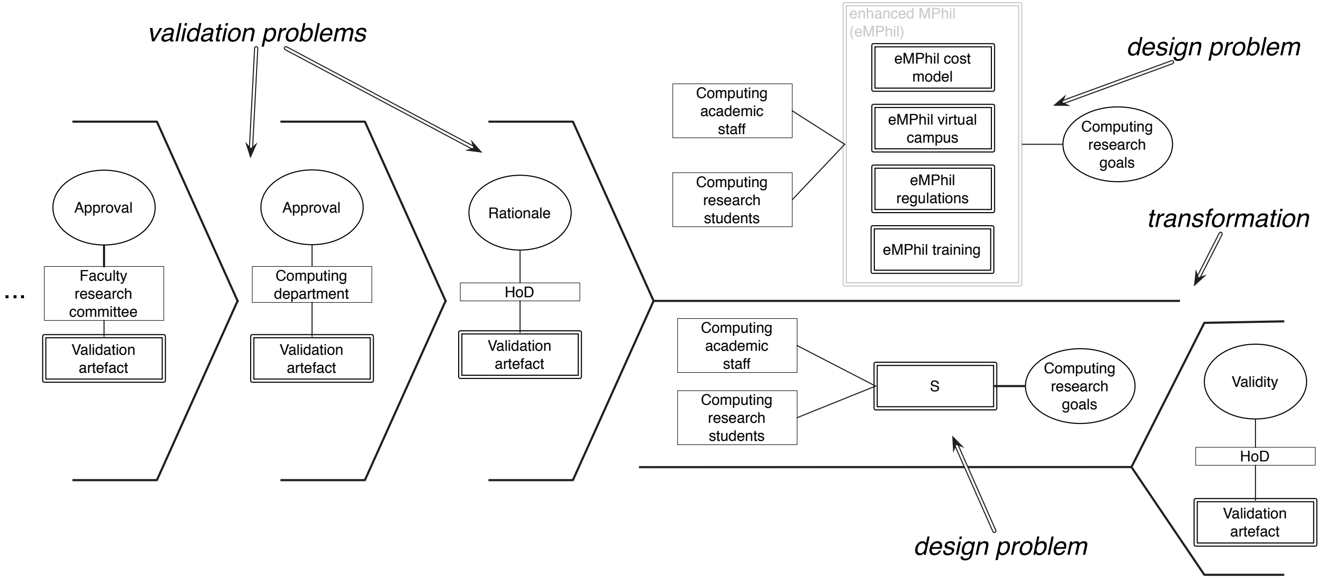


Figure 6. Early design steps

an empty conclusion problem—the start of all POE Design explorations), with a problem validator identified as the Head of the Computing Department (HoD). The HoD set the strategic goals which constituted the initial requirement description, and led to the inclusion of Computing academic staff and research students as a first approximation for the problem context. This initial problem exploration was coupled with the solution of the associated validation problem, consisting in making sure that the HoD's strategic intent was understood correctly by the problem solver.

The next transformation (top of Figure 6) captures an early solution exploration activity, in which a candidate solution architecture is starting to emerge, that of a new research degree, an MPhil, to be delivered in part-time mode at a distance. Note the validation problems to the left. The initial buy-in for the new degree was sought from the HoD, as the person in charge of releasing resources for the project, and with whom the rationale for the proposed solution was discussed. Approval from the HoD then triggered a comprehensive approval process throughout the organisation, reflecting its power structure (each validation problem concerns stake-holders at different management levels). Downside risks at these point were very high, with assurance taking precedence and greatly influencing the design.

Subsequent design alternated between further problem and solution explorations, and related validation, as illustrated in Figure 8, which provides a snapshot of the design tree after the first 10 months' development from the developer team perspective, assuming both problem and solution finder roles. Transformations labelled A and B at the bottom of the figure correspond to the early steps we have just described. From the initial solution architecture, a number of sub-problems were then identified (transformation C) each addressing complementary aspects of the solution, such as the design of its technological infrastructure, a related cost model, a programme of user induction and training, a system of monitoring and evaluation, etc. Each sub-problem was then taken forward through further transformations and related stake-holder validation, with the design problems at the top representing either solved sub-problems or open problems in the process of being addressed.

Note that some of the steps introduce sub-problems, which lead to branches in the design tree. This happens when a number of solution components have been identified, each to be designed, together with their architectural relations and mutual constraints. The POE transformation which generates them, called *solution expansion*, generates appropriate sub-problems for each to-be-designed component, based on such architectural knowledge. Each sub-problem then becomes the root of a (sub-) design tree.

4.2.3 Fractal Problem Validation

As introduced in Section 3, validation problems are problems too, and so their solution can be arrived at through a problem solving process, and hence (should they have solution) solvable in our POE framework, i.e., they should be treated as any other problem, with problem finder exploring the problem, obtaining problem validation, and so on. In the augmented notation, the obvious place for the validation problem development is in the extension to the horizontal bar; see Figure 7. However, such diagrams quickly become unwieldy, and a more pragmatic approach was necessary in which the validation problem development was placed in a separate file, with indicators (again, Figure 7, on the left) for the state of each validation problem and hyperlinks used for easy access to the embedded validation problem development. It became apparent that the indicators formed a useful proxy for developmental risk associated with an unsolved validation problem, that risk being associated with the progress made in the solution of the main problem as opposed to the validation problem. We used a simple semaphore system for the risk indicators. Given the lack of tool support, this was deemed a simple, but useful tool from a project management perspective; of course, a more accurate estimation of risk would have required more sophisticated tools. Figure 7 gives an intuition of the meaning of the risk indicator: to the right is the equivalent fully expanded validation problem.

4.3 Early evaluation

The experience on the project so far has been very encouraging, and has clearly indicated that the conceptual tools offered by POE, including assurance-driven design were able to cater for all relevant aspects of the project. Design forests provide a powerful summary of the development, with all critical decision points clearly exposed, and all sub-problems (solved and unresolved) and their relation clearly identified. The risk indicators, despite their lack of sophistication, were considered very useful in signposting critical parts of the development. The notation was also considered an effective communication tool: its relative simplicity and abstraction allowed even non technical stake-holders, like senior managers in academic and academic-related units, to grasp the essence of the project with very little explanation required. The inclusion of validation problems within the development tree, with the explicit acknowledgement of all relevant stake-holders, was also considered a valuable tool to gauge the criticality of each design step, as well as to focus attention on the aspects of the problem of significance to each stake-holder. For instance, the high criticality of initial approval process is evidenced by the large set of validation problems in the early stages of development, in which the validation effort largely outweighed the effort to produce an initial outline for the solution, but greatly reduced the risk of the programme not to be deemed viable by management later on.

5 Related Work

Work on assurance cases is found in the area of dependability, from which two main structured notations for expressing safety cases have emerged. One is the goal-structuring notation (GSN) [21], a graphical argumentation notation which allows the representation of individual elements of a safety argument and their relations. Elements include: goals (used to represent requirements and claims about the system), context (used to represent the rationale for the approach and the context in which goals are stated), solutions (used to give evidence of goal satisfaction) and strategies (the approach used to identify sub-goals). The other, is Adelard's Claim-Argument-Evidence (ASCAD) approach [22], which is based on Toulmins work on argumentation and includes: claims (same as Toulmin's claims), evidence (same as Toulmin's grounds) and argument (combination of Toulmin's warrant and backing). More recently, Habli and Kelly [23] have also suggested ways in which product and process evidence could be combined in GSN assurance cases. One of the difficulties of these approaches is that they were not conceived to provide an integrated approach to safety development and, by and large, use artifacts and processes which may parallel but not integrate with software development. Instead, a main aim of our work is to allow for the efficient co-design of both software and assurance case based on artefacts and processes which are common to both. Some very recent work by Strunk and Knight [24] proposes Assurance Based Development (ABD) in which a safety-critical system and its assurance case are developed in parallel through the combined used of Problem Frames [7] and GSN. Although this work shares some of our goals, it is still rather preliminary for a meaningful comparison with POE.

A more mature process model, which shares something with POE, is the CHOAS model and lifecycle of Raccoon [25]. In this model fractal invocations of problem solving processes are combined to provide a rich model of software development, which is then used as the basis for a critical review of software engineering, of its processes and its practices. Raccoon's review leads to the conclusion that neither separately nor together do top-down or bottom-up developments tell the whole story; hence, a 'middle out theory' is proposed, based on the work that developers do to link high level project issues to, essentially, code structures. It is an attractive theory, and we wish to explore the ways in which fractal invocation in POE and assurance-driven design satisfy the criteria laid down for it.

6 Discussion and conclusion

The POE notion of problem suggests a separation of context, requirement and solution, with explicit descriptions of what is given, what is required and what is designed. This improves the traceability of artefacts and their relation, as well as exposing the assumptions upon which they are based to scrutiny and validation. That all descriptions are generated through problem transformation forces the inclusion of an explicit justification that such assumptions are realistic

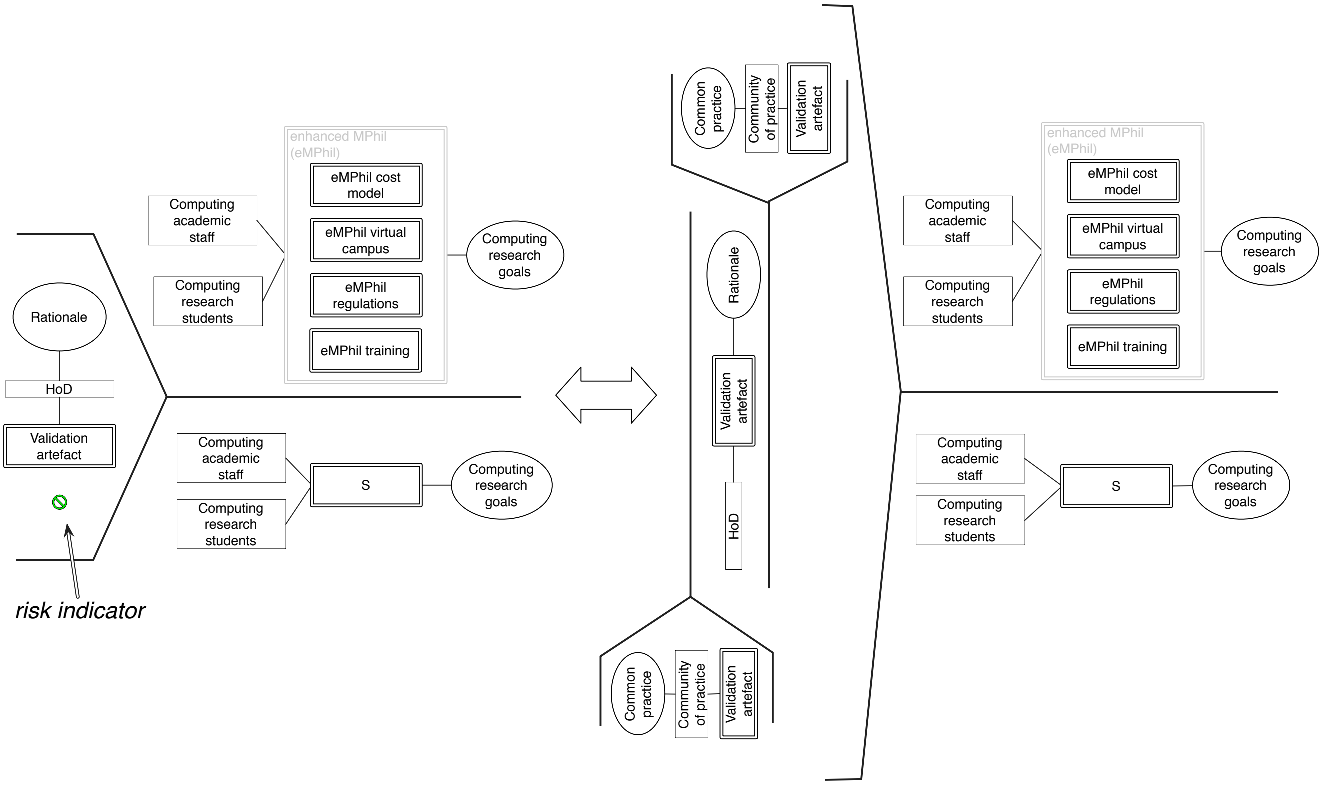


Figure 7. Risk indicator and its 'fractal' validation problem

and reasonable. In particular, requirements are justified as valid, are fully traceable with respect to the designed system (and *vice versa*), and evidence of their satisfaction is provided by the adequacy argument of a completed POE development tree.

We have shown (a) how (partial) problem and solution validation can be used to manage developmental risk and (b) how an assurance arguments can be constructed alongside the development of a product. Developmental risks arise from tentative transformation which are not completely justified: in such cases concerns can be stated as suspended justification obligations to be discharged later on in the process. This adds the flexibility of trying out solutions, while still retaining the rigour of development and clearly identifying points where backtracking may occur.

Although other approaches provide a focus on an assurance argument, the possibility of having the assurance argument *drive* development is an option that appears unique to ADD and POE.

Finally, POE defines a clear formal structure in which the various elements of evidence fit, that is whether they are associated with the distinguished parts of a development problem or the justifications of the transformation applied to solve it. This provides a fundamental clarification of the type of evidence provided and reasoning applied. Moreover, that the form of justification is not prescribed under POE signifies that all required forms of reasoning can be accommodated, from deductive to judgemental, within a single development.

Acknowledgments

We acknowledge the financial support of IBM, under the Eclipse Innovation Grants, and of SE Validation Limited. Our thanks go to Derek Mannering at General Dynamics UK, Lucy Hunt at Getronics plc, Jens Jorgensen and Simon Tjell of Aarhus University, Andrés Silva of the University of Madrid, Colin Brain of SE Validation Ltd, Anthony Finkelstein at UCL (who suggested the discussion of "Have we done enough?"), and John Knight at UVA. L. B. S. Raccoon has read all of our work and made truly insightful comments. Finally, thanks go to our many colleagues in the Computing Department at The Open University, particularly Michael Jackson.

References

[1] Jon G. Hall and Lucia Rapanotti. Assurance-driven design. In *Proceedings of the Third International Conference on Software Engineering Advances (ICSEA 2008)*. Published

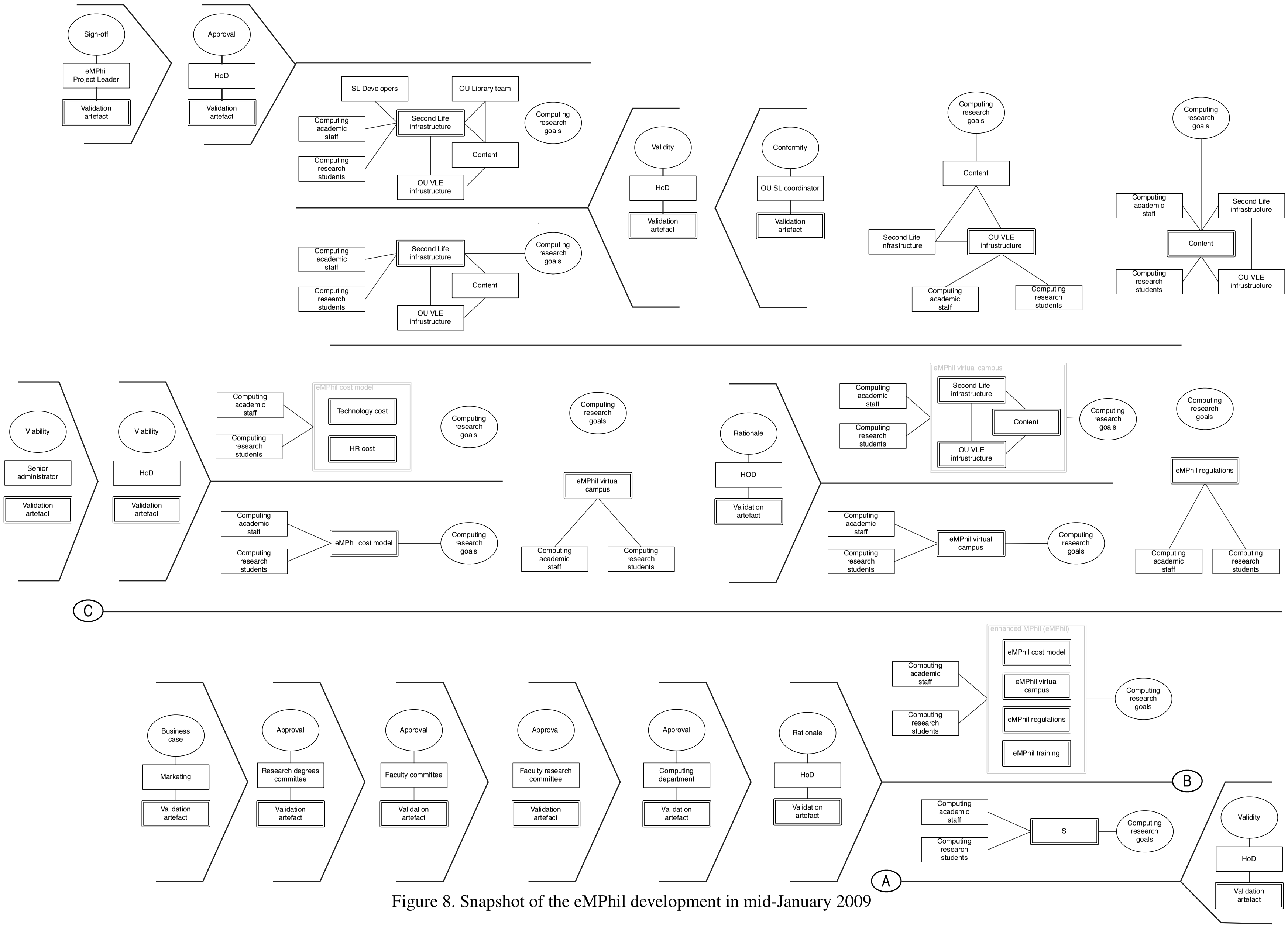


Figure 8. Snapshot of the eMPhil development in mid-January 2009

by the IEEE Computer Society, 2008. Also available as Open University Computing Department Technical Report #2007/15.

[2] Engineering Council of South Africa Standards and Procedures System Definition of Terms to Support the ECSA Standards and Procedures System.

[3] Jon G. Hall, Lucia Rapanotti, and Michael Jackson. Problem oriented software engineering: A design-theoretic framework for software engineering. In *Proceedings of 5th IEEE International Conference on Software Engineering and Formal Methods*, pages 15–24. IEEE Computer Society Press, 2007. doi:10.1109/SEFM.2007.29.

[4] Jon G. Hall, Lucia Rapanotti, and Michael Jackson. Problem-oriented software engineering: solving the package router control problem. *IEEE Trans. Software Eng.*, 2008. doi:10.1109/TSE.2007.70769.

[5] M. E. Szabo, editor. *Gentzen, G.: The Collected Papers of Gerhard Gentzen*. Amsterdam, Netherlands: North-Holland, 1969.

[6] Daniel Jackson. *Software Abstractions: Logic, Language, and Analysis*. MIT Press, Cambridge, MA, 2006.

[7] Michael A. Jackson. *Problem Frames: Analyzing and Structuring Software Development Problems*. Addison-Wesley Publishing Company, 1st edition, 2001.

[8] Derek Mannering, Jon G. Hall, and Lucia Rapanotti. Towards normal design for safety-critical systems. In M. B. Dwyer and A. Lopes, editors, *Proceedings of ETAPS Fundamental Approaches to Software Engineering (FASE) '07*, volume 4422 of *Lecture Notes in Computer Science*, pages 398–411. Springer Verlag Berlin Heidelberg, 2007.

[9] Jon G. Hall, Derek Mannering, and Lucia Rapanotti. Arguing safety with problem oriented software engineering. In *10th IEEE International Symposium on High Assurance System Engineering (HASE)*, Dallas, Texas, 2007.

[10] Jon G. Hall and Lucia Rapanotti. The discipline of natural design. In *Proceedings of the Design Research Society Conference 2008*. Design Research Society, 2008.

[11] Derek Mannering, Jon G. Hall, and Lucia Rapanotti. Safety process improvement with POSE & Alloy. In Francesca Saglietti and Norbert Oster, editors, *Computer Safety, Reliability and Security (SAFECOMP 2007)*, volume 4680 of *Lecture Notes in Computer Science*, pages 252–257, Nuremberg, Germany, September 2007. Springer-Verlag.

[12] OMG. Unified Modeling Language (UML), version 2.0. http://www.omg.org/technology/documents/formal/uml.htm. Last checked: May 2009.

[13] Donald Firesmith. *The Method Framework for Engineering System Architectures*. CRC Press, 2008.

[14] James P. Womack and Daniel T. Jones. *Lean Thinking – Banish Waste and Create Wealth in Your Corporation*. Simon and Schuster, 1996.

[15] Kenneth Falconer. *Fractal Geometry: Mathematical Foundations and Applications*. Wiley-Blackwell, 2nd edition, 2003.

[16] Alistair Cockburn. *Writing Effective Use Cases*. Addison-Wesley, 2001.

[17] Lucia Rapanotti and Jon G. Hall. Designing an online part-time master of philosophy with problem oriented engineering. In *Proceedings of the Fourth International Conference on Internet and Web Applications and Services*, Venice, Italy, May 24-28 2009. IEEE Press.

[18] Lucia Rapanotti and Jon G. Hall. Problem oriented engineering in action: experience from the frontline of postgraduate education. Technical Report TR2008/16, The Open University, 2008.

[19] Lucia Rapanotti, Leonor M. Barroca, Maria Vargas-Vera, and Shailey Minocha. deepthink: a second life campus for part-time research students at a distance. Technical Report TR2009/1, The Open University, 2009.

[20] UK GRAD, Joint Skills Statement of Skills Training Requirements. http://www.grad.ac.uk/jss/ Last checked: May 2009.

[21] Tim Kelly. A systematic approach to safety case management. In *Proceedings SAE 2004 World Congress*, Detroit, US, 2004.

[22] R. Bloomfield, P. Bishop, C. Jones, and P. Froome. *ASCAD - Adelard Safety Case Development Manual*, 1998.

[23] I. Habli and T. Kelly. Achieving integrated process and product safety arguments. In *Proceedings of 15th Safety Critical Systems Symposium (SSS'07)*. Springer, 2007.

[24] Elisabeth A. Strunk and John C. Knight. The essential synthesis of problem frames and assurance cases. *Expert Systems*, 25(1):9–27, 2008.

[25] L. B. S. Raccoon. The Chaos model and the Chaos cycle. *SIGSOFT Softw. Eng. Notes*, 20(1):55–66, 1995.

Reconfigurable Service-Oriented Architecture for Autonomic Computing

Radu Calinescu
Computing Laboratory, University of Oxford
Wolfson Building, Parks Road, Oxford OX1 3QD, UK
Radu.Calinescu@comlab.ox.ac.uk

Abstract

Almost a decade has passed since the objectives and benefits of autonomic computing were stated, yet even the latest system designs and deployments exhibit only limited and isolated elements of autonomic functionality. In previous work, we identified several of the key challenges behind this delay in the adoption of autonomic solutions, and proposed a generic framework for the development of autonomic computing systems that overcomes these challenges. In this article, we describe how existing technologies and standards can be used to realise our autonomic computing framework, and present its implementation as a service-oriented architecture. We show how this implementation employs a combination of automated code generation, model-based and object-oriented development techniques to ensure that the framework can be used to add autonomic capabilities to systems whose characteristics are unknown until runtime. We then use our framework to develop two autonomic solutions for the allocation of server capacity to services of different priorities and variable workloads, thus illustrating its application in the context of a typical data-centre resource management problem.

Keywords: autonomic computing, self-* system, service-oriented architecture, model-driven development, reconfigurable system

1. Introduction

The onset of a *digital economy* led to revolutionary transformations to the way in which Information and Communication Technologies (ICT) are used to conduct business and research and to provide services in all sectors of the society [2, 3]. The ability to accomplish more, faster and on a broader scale through expert use of ICT is at the core of today's scientific discoveries, newly emerged services and everyday life. Due to unprecedented advances in ICT, business needs are attended to by ever more complex and feature-rich systems and systems of systems [4].

Autonomic computing represents a powerful approach to managing the spiralling ICT complexity brought by these developments, by reducing the level of expertise required from the end users of ICT systems, and leveraging the rich capabilities of complex ICT components. Formally launched less than a decade ago [5], autonomic computing proposes that the demanding tasks of configuring, optimising, repairing and protecting complex ICT systems are delegated to the systems themselves [6]. Based on a set of high-level objectives (or *policies*), autonomic systems are intended to “manage themselves according to an administrator's goals” [7].

Following several years of intense research, we now have a good understanding of what autonomic systems should look like [6, 7, 8, 9, 10] and what best practices to follow in building them [11, 12, 13, 14]. This significant progress is to a great extent a by-product of the effort that went into the development of successful autonomic solutions addressing specific management tasks in real-world applications [15, 16, 17, 18, 19, 20].

While these developments demonstrate the feasibility and advantages of the autonomic computing approach to complexity management, autonomic functionality is far from ubiquitous in today's ICT systems. In previous work, we used insights from the development of a commercial autonomic system for the management of data-centre resources [15] to identify key challenges in the development of autonomic systems [11], including:

- The lack of standardisation in ICT resource interfaces. Despite an increasing trend to add management interfaces to new ICT components and devices, and to make existing interfaces public, autonomic system development is hindered by the broad diversity of architectures and technologies these interfaces are based upon.

- The tendency to hardcode ICT resource metadata within the control component of the autonomic system. Management frameworks are often intended for handling particular types of resources, and the parameters of these resource types are hardcoded in the control element of the system. With careful design, complex systems consisting of supported resources can be successfully managed; however, adding in support for additional types of resources cannot be achieved in a cost-effective way.

- The high scalability expectations. As simple, small ICT systems are easy to manage by low-skilled human operators, autonomic solutions are required in areas where the systems to manage are complex and comprise large numbers of resources.

Based on best practices devised while investigating these challenges [11], we then proposed a generic autonomic framework for the effective development of autonomic solutions in [21, 22] and briefly described its implementation as a service-oriented architecture (SOA) in [1]. This article represents an extended version of [1]. As such, the article provides additional information about our generic autonomic framework and the way in which it addresses the challenges mentioned above. Also, the article provides a significantly enhanced description of the framework implementation and of the combination of automated code generation, model-based and object-oriented development techniques employed by this implementation. Finally, we present a new autonomic solution for the typical server capacity allocation problem from [1], thus additionally illustrating how the framework can be used to support utility-function autonomic computing policies (i.e., policies that require adjusting the configurable parameters of a system so as to maximise the value of a user-specified *utility function* [23]).

The remainder of the article is organised as follows. Section 2 provides an overview of the generic autonomic computing framework, and examines how existing standards, technologies and tools can be used for its practical realisation. Section 3 presents the implementation of the autonomic architecture proposed by our framework as a service-oriented architecture, a solution chosen in order to take advantage of web service technology benefits such as platform independence, loose coupling and security support [24]. In Section 4, a case study involving the allocation of server capacity to services of different priorities and variable workloads is used to illustrate the application of the framework. Section 5 reviews related work in the areas of autonomic systems development out of legacy resources, model-driven development of autonomic systems and autonomic computing expression languages. Finally, Section 6 summarises our results and discusses a number of further work directions.

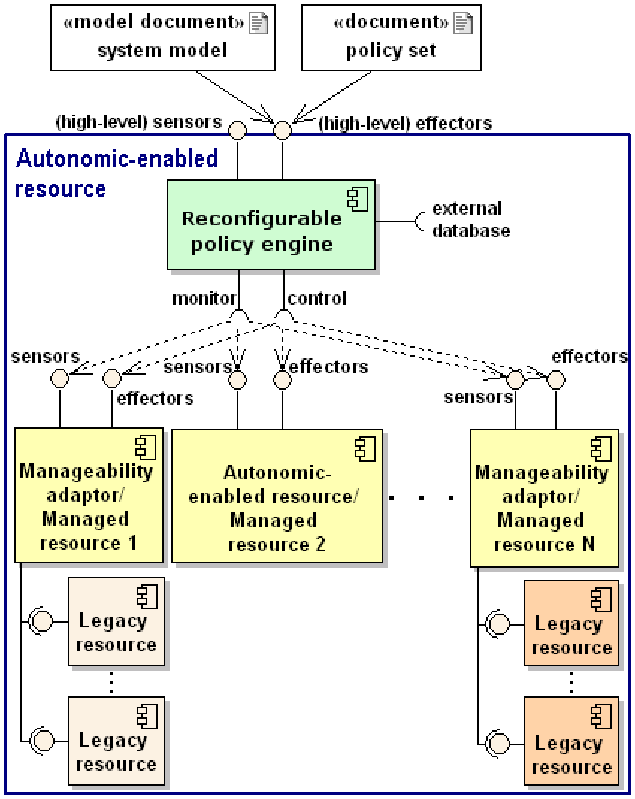


Figure 1. UML component diagram of the general-purpose autonomic architecture

2. Overview of the generic autonomic framework

Figure 1 depicts the general-purpose autonomic architecture used by our framework. Originally introduced in [21] and further developed in [22], this architecture builds on the recent developments mentioned in the introductory section, and extends the author's previous work on the policy-based management of data-centre resources [15].

The core component of the architecture is a reconfigurable policy engine that organises a heterogeneous collection of legacy ICT resources (i.e., resources not designed to support management by the policy engine) and autonomic-enabled resources into a self-managing system. In order to make autonomic solution development cost-effective, the policy engine can be configured to handle resources whose types are unknown during its implementation and deployment.

The rest of this section describes the components of the architecture and how they enable the runtime reconfiguration of the policy engine. Existing standards, technologies and tools are suggested that can be employed to realise instances of these components.

2.1. Managed resources

The legacy ICT resources whose complexity can be managed through their integration into instances of the architecture include:

- physical and application servers, software applications;
- virtualisation environments and virtual machines;
- ICT devices such as switches and load balancers;
- factory automation equipment and robotic systems;
- household devices such as home safety and security devices.

The autonomic-enabled resources in the self-managing system are either typical ICT resources that were specifically designed to expose *sensors* and *effectors* interfaces allowing their direct inter-operation with the policy engine, or other instances of the architecture. As illustrated in Figure 1, the latter option is possible because the policy engine is exposing the entire system as an atomic ICT resource through *high-level sensors* and *high-level effectors*.

The high-level sensors expose to the outside world:

- The state of the policy engine itself, namely the current values of the engine parameters; for our implementation of the policy engine, these parameters are presented in Section 3.2.
- An overall view of the system state. Note that because the purpose of the high-level sensors is to facilitate the integration of the autonomic system as a component into a larger system, this view will typically—but not necessarily —represent a *summary* of the system state. Possible examples of such a summary include the average load of the servers within the system, the mean response time for a set of web applications or the failure rate of the system components. The precise nature of the system view presented by the high-level sensors is defined by special policies supplied to the policy engine, as described later in this section.

Likewise, the high-level effectors expose the configuration parameters of the engine (these parameters are described in Section 3.2), and any system-wide configuration parameters specified by the user-provided policy set.

2.2. Manageability adaptors

As recommended by IBM's architectural blueprint for autonomic computing [13], standardised adaptors are used to expose the *manageability* of all types of legacy ICT resources in a uniform way, through sensor and effector interfaces. These two types of interfaces enable the policy engine to access the state of the legacy resources and to configure their parameters, respectively—all without any modification to the managed resources. For efficiency reasons, the sensors should support both explicit reading of specific state information and, whenever possible, a state-change notification mechanism that the policy engine can subscribe to.

Note that the manageability adaptor interfaces for any instance of our architecture are fully defined by the system model used to configure the policy engine. This makes possible the use of model-based development techniques and tools for the semi-automatic generation of the manageability adaptors. By carefully selecting the technology used to "encode" the system model, off-the-shelf tools can be employed for this purpose—this is illustrated in Section 3, where XML system models are used for the configuration of the policy engine.

Another good approach to implementing the manageability adaptors is based on the OASIS Web Services Distributed Management (WSDM) standard. The Management Using Web Services (MUWS) component of WSDM [25] defines a web service architecture enabling the management of generic distributed resources. The MUWS specification describes a standard way in which manageable resources can expose their *capabilities*, and defines a number of built-in capabilities that resources should provide (e.g., *ResourceId*, *Description* and *Version*). Resource-specific capabilities can be provided and listed as elements of the *ManageabilityCharacteristics* built-in capability. The MUWS standard specifies ways for accessing resource capabilities by means of web services, and requires that a "resource properties document" XML schema is provided as a basic model of the managed resources. An integrated development environment for the implementation of WSDM-compliant interfaces is currently available from IBM [26].

2.3. System model

Information about the system under the control of the policy engine—including details about its parameters—is provided by a *system model* that is supplied to the engine at run time. This model represents a specification of all resources to be managed and of their relevant *properties*. Note that the parameters of a system resource (e.g., the CPU capacity of a server, or the name of a process running on this server) are termed "properties" throughout most of the article in order to match the terminology proposed by the

WSDM standard [25].

As the engine can always be reconfigured using new versions of the model, resources and resource properties not referred to in the policies need not be specified. To allow the use of appropriate operators in autonomic computing policies and to reduce the amount of work by the policy engine, the model should provide details about each resource property it defines, including:

- the data type of the property;
- whether the property is read-only or can be modified by the policy engine;
- if the property has a constant value or it changes over time;
- if the policy engine can request to be notified about changes in the property value.

Because the policy engine needs to handle new system models at runtime, two further requirements must be satisfied by these models. The first requirement is that all system models are instances of the same meta-model, and the second that they are expressed in a format that the engine can use to generate automatically any components it needs to inter-operate with the manageability adaptors (e.g., classes for new resource types or manageability adaptor proxies).

Several standards and technologies are good candidates for the representation of the system model:

- Microsoft's System Definition Model (SDM) is a meta-model used to create models of distributed systems [27] with a high degree of detail. The ongoing Dynamic Systems Initiative programme [28] intends to use these complex models as enabling elements in the development of manageable systems that exhibit elements of autonomic behaviour. Given its complexity, the SDM meta-model is less suited for use in conjunction with the reconfigurable policy engine employed by our autonomic architecture.
- The WSDM/MUWS standard [25] uses the WS-Resource Metadata Descriptor framework to describe the metadata for a resource manageability endpoint. This allows the specification of the properties of resource state variables and parameters, as well as the definition of resource relationships and *operable collections* (i.e., set of resources with aggregated state and operations).
- The Service Modeling Language (SML) specification put together by a consortium of leading ICT companies [29] can be used to model complex ICT resources based on a philosophy similar to that underlying the design of our autonomic architecture. When SML is adopted as a W3C standard and an SML development toolset becomes available, the use of SML models for the configuration of the policy engine will become a compelling option.
- The Managed Resource Document (MRD) used by version 1.1 of IBM's Policy Management for Autonomic Computing (PMAC) framework, and the combination of web services and autonomic computing standard specifications that version 1.2 of PMAC uses are further examples of managed system models [30].

Finally, in order to make the development of these system models and of the autonomic solutions they underpin cost-effective, their elements need to be drawn from resource definition repositories built around domain-specific ICT ontologies [11]. This enables the reuse and sharing of manageability adaptors and policies across autonomic solutions from the same application domain, therefore leveraging the advantages of ontology-based modelling in the realm of autonomic computing, as emphasised in [31] and demonstrated successfully by [32].

2.4. Autonomic computing policies

Our policy model (Figure 2) extends the policy paradigm in [15, 30, 33, 34] based on best practices proposed in [11]. The abstract $Policy$ type at the root of the policy class hierarchy comprises three elements that are common to all policy types:

- The *policy scope* specifies the resources to which the policy applies, and takes the form of a set of "resource group" expressions. Each such expression is specified as a filter applied to a resource type supported by the policy engine. For example, given a cluster of servers, resource group expressions can be used to select all processes running on these servers and whose name matches a regular expression *regex* (i.e., `process.name` $=\sim$ *regex*) and/or all servers whose CPU utilisation exceeds 75% (i.e., `server.cpuUtilisation > 75%`).
- A *policy value* specified as an arithmetic expression is associated with each policy, and in the presence of conflicting/competing policies, higher-priority policies are realised at the expense of lower priority ones. In its simplest form, a policy value is an integer number.
- The *policy condition* is a Boolean expression used to specify the circumstances in which the policy engine is required to perform an action. This expression can use as parameters properties of the system resources in the policy scope, or built-in system variables such as `time`. For example, the policy condition '`time.hour>=9 &`

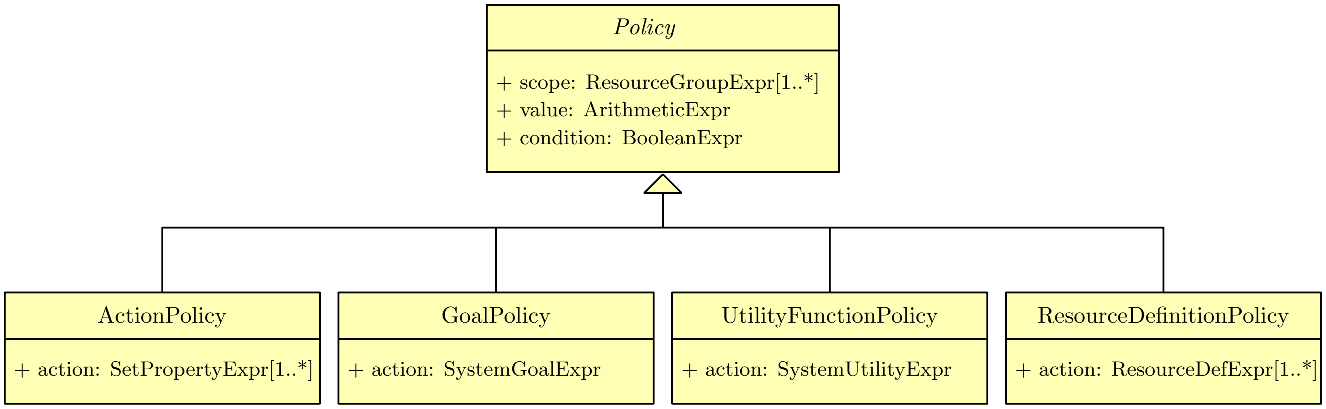


Figure 2. Policy model

`time.hour<=17`' specifies that the associated policy action should be performed between 9am and 5pm.

As illustrated in Figure 2, the abstract *Policy* class is specialised by four concrete classes of policies. These policy classes are associated with the *action*, *goal* and *utility function* policy types defined in [23, 35, 36], and with a *resource-definition* policy type that specifies how the policy engine should expose the managed resources through its high-level sensor and effector interfaces. The difference among these policy types is in the way in which they specify the fourth element of an autonomic computing policy, namely its *action*:

- The action element of an *action policy*[1] specifies new values for one or several properties of the resources within the scope of the policy. For this reason, such actions are encoded as sequences of assignment expressions of the form *`resource_property = expression`*.

- The action element of a *goal policy* is a Boolean expression that depends on the properties of the resources in the policy scope. Given a goal policy, the policy engine should adjust the modifiable properties of the resources in the policy scope in order to ensure that this Boolean expression evaluates to `true` at all times. For example, a goal policy may be specified that requests the policy engine to maintain the response time of all services in the policy scope below 1500ms—'`MAX(service, service.responseTime)<1500`'.[2]

- The action element of a *utility-function policy* specifies a "utility function" that associates a numerical value with each state of the resources in the policy scope (i.e., with the values of their properties). The policy engine is required to adjust the modifiable properties of these resources in order to bring them into a state that corresponds to the maximimum value of the utility function that is attainable. Utility-function policies are described in more detail in the context of the case study in Section 4.2.

- The action element of a *resource-definition policy* defines new types of resources that the policy engine is required to synthesise. The names and properties of these new resource types are fully specified by the action element of the resource-definition policy, and the policy engine is required to synthesise the software components for these resources dynamically. Presenting the semantics and implementation of resource-definition policies, and their role in the development of autonomic systems of systems is beyond the scope of this article—this information is available instead in a related publication by the author [58].

As described so far in this section, the four elements of a policy are specified in terms of expressions of appropriate type, and the ability to apply a rich set of operators and functions to the resource properties used in these expressions is key to supporting the types of policies in Figure 2. Accordingly, the policy language should include:

- an extensive set of operators for the manipulation of primitive types like the one provided by IBM's Autonomic Computing Expression Language [33];

[1] For historical reasons (in the early days of autonomic computing, action policies were the only type of autonomic computing policies), the term action is used to denote a component of autonomic computing policies, as well as a type of such policy. The meaning should be obvious from the context.

[2] Some of the techniques that the policy engine can employ to implement goal and utility policies are described in Section 4.2.

- regular expression and time operators similar to those implemented by Microsoft's Windows System Resource Manager [37];
- functions calculating average/minimum/maximum resource property values over a time interval and/or across a resource set like the built-in operators of the commercial policy engine in [15].

Additionally, a number of operators from areas such as formal specification [38] and formal quantitative analysis [39] are required to support or simplify the encoding of the four policy elements. These include *set comprehension* and *transitive closure* [38], for specifying the "resource group" expressions in the scope of policies; existential and universal quantification operators, to support the specification of policy conditions; and operators varying from ordinary assignments to choice, scheduling, linear programming and other optimisation operators for defining the actions of goal, utility-function and resource-definition policies.

2.5. Reconfigurable policy engine

The internal architecture of the reconfigurable policy engine (Figure 3) is dictated by the types of policies it implements and by its ability to handle ICT resources whose characteristics are supplied to the engine at runtime. A "coordinator" module is employing the components described below to implement the closed control loop of an autonomic system.

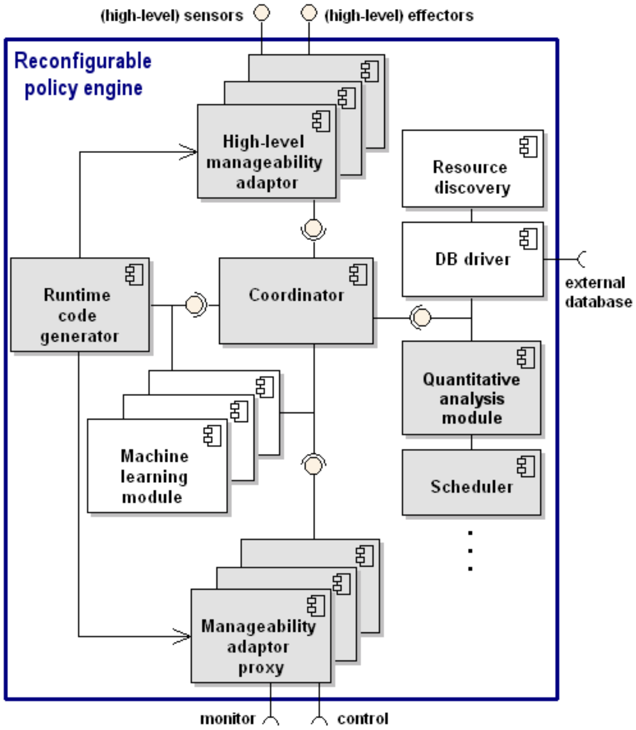


Figure 3. Architecture of the policy engine. The shaded components are implemented by the prototype described in Section 3.

Runtime code generator This component generates the necessary interfaces when the policy engine is configured to manage new types of resources or supplied with new "resource definition" policies. When a new system model is used to reconfigure the policy engine, *manageability adaptor proxies* are generated that allow the engine to interoperate with the manageability adaptors for the resource types specified in the system model. Likewise, when "resource definition" policies are set up that specify new ways in which the policy engine should expose the ICT resources it manages, *high-level manageability adaptors* need to be generated.

Manageability adaptor proxies These modules are thin interfaces allowing the policy engine to communicate with the autonomic-enabled resources and the manageability adaptors for the legacy resources in the system.

High-level manageability adaptors These elements are used to expose the system state and configuration in a format that allows its integration within another instance of the general-purpose autonomic architecture. The exposed system characteristics include the state and configuration of the policy engine itself (e.g., system model, policy set and monitoring period), as well as any characteristics of the managed resources that are specified by 'resource definition" policies implemented by the engine.

Scheduler This module is used to support the various operators appearing in policy actions for the goal and "utility function" policies handled by the policy engine, examples of which are provided in Section 4.

Resource discovery This component is used to locate the resources to be managed by the policy engine. The use of a technique such as the adaptive resource discovery described in [40] is recommended, although simpler approaches may be suitable for some use cases.

Database driver This module is used to maintain policy engine data such as historical resource property values in an external persistent storage.

Machine learning modules The ability to implement goal and "utility function" policies is key to the effective management of complexity within an autonomic system. However, this requires the policy engine to possess in-depth knowledge about the behavioural characteristics of the managed system that should not (and often cannot) come from the system administrator. We are proposing that machine learning techniques [41] are employed by a set of policy engine modules to generate a behavioural (or *operational* [35]) model of the managed ICT resources based on sensor data and inside policy engine information. The usefulness of a *Modeler* component in autonomic systems that support utility functions is mentioned in [23], although the authors are not specific about the learning algorithms that such a component might use.

Quantitative analysis module This component enables the policy engine to take full advantage of a quantitative behavioural model that may be provided as part of the system model in Figure 1 or, in the future, built by its machine learning modules. The use of this module to support the implementation of a powerful class of utility-function policies represents the subject of a forthcoming paper [42].

3. Implementation

Two major choices influence the way in which an instance of the architecture in Figure 1 is realised: the technology used to represent the system model; and the technology chosen for the implementation of the policy engine components. This section describes how we made these choices for a prototype implementation of the architecture and gives prototype implementation details.

3.1. System model

For our prototype implementation, we chose to represent system models as plain XML documents that are instances of a pre-defined meta-model encoded as an XML schema. This choice that disregards some of the better suited modelling technologies discussed in Section 2.3 (e.g., [25, 27, 29]) was motivated by the availability of numerous “off-the-shelf” tools for the manipulation of XML documents and XML schemas that are largely lacking for the other technologies. In particular, by using existing XSLT engines and XML-based code generators we shortened the prototype development time and avoided the need to implement bespoke components for this functionality.

As illustrated by the UML class diagram in Figure 4, our meta-model specifies a managed system as a named set of resource definitions. Each resource definition (i.e., *resourceDefinition* in the UML diagram) comprises a unique identifier *ID*, a description and a set of resource properties with their characteristics. A resource property has a data type (i.e., *propertyDataType*), and is associated a unique *ID* and the metadata repository URL where its definition is available. Several other property characteristics are defined in the meta-model:

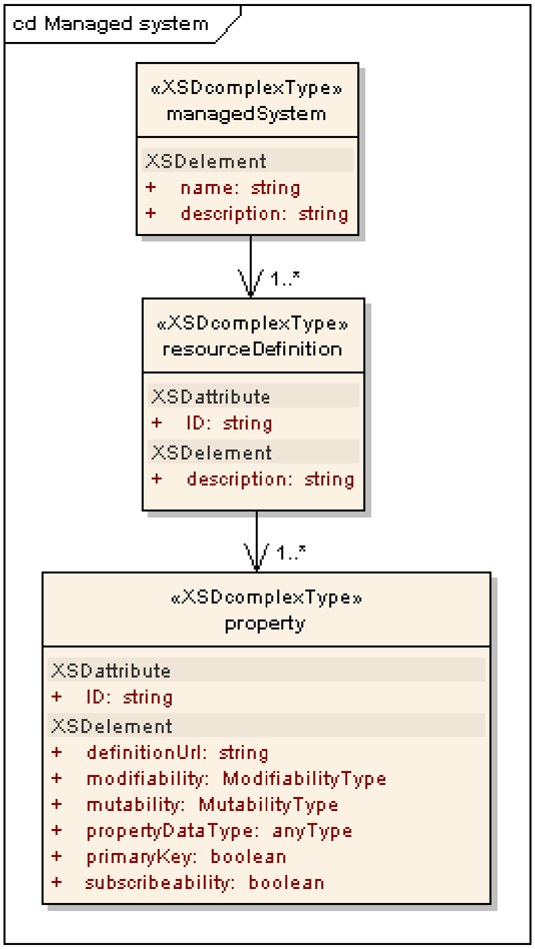


Figure 4. Meta-model of a managed system

- *modifiability*—taken from the WS-ResourceMetadata-Descriptor (WS-RMD) 1.0 specification [43], specifies if the property is “read-only” or “read-write”;

- *mutability*—the WS-RMD MutabilityType [43] specifies if the property is “constant”, “mutable” or “appendable”;

- *primaryKey*—indicates whether the property is part of the property set used to identify a resource instance among all resource instances of the same type.

- *subscribeability*—specifies whether a client such as the policy engine can subscribe to receive notifications when the value of this property changes;

3.2. Policy engine

The generality of the autonomic architecture described in Section 2 allows the implementation of the reconfigurable policy engine using different technologies, e.g., as a software agent running on a data-centre server or a physical device incorporated into an industrial robotic system.

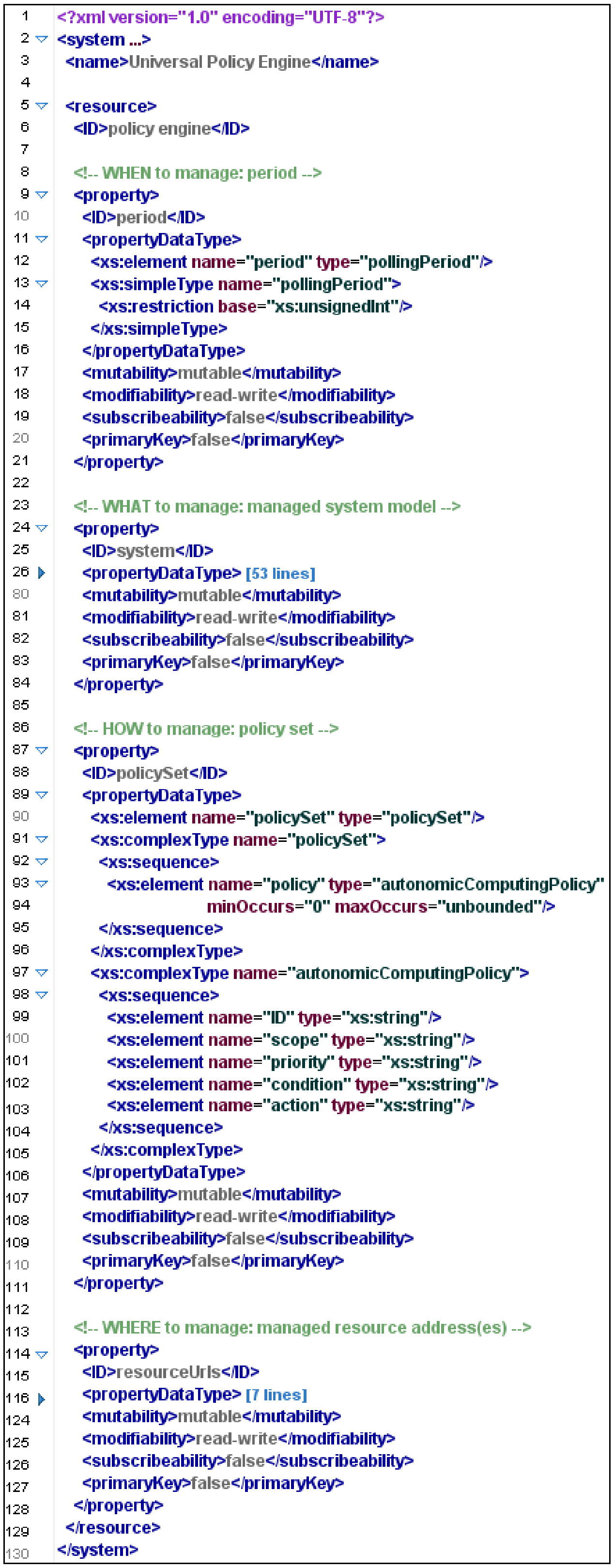
Our prototype policy engine and the manageability adaptors enabling its interoperation with legacy resources were implemented as web services in order to leverage the platform independence, loose coupling and security features of this technology. The runtime reconfiguration of the policy engine necessitated the extensive use of techniques available only in an object-oriented (OO) environment:

- Dynamic generation of data types (i.e., classes) was required to support new types of resources when the policy engine was reconfigured by means of a new system model.
- Runtime generation of web service proxies was required to enable the policy engine to interoperate with new, resource-specific manageability adaptors.
- *Reflection* (i.e., an object-oriented programming technique that allows the runtime discovery and creation of objects based on their metadata [44]) was heavily used to access the values of the resource properties, both to read their values once the policy engine obtained them from the manageability adaptors and to set new values for the modifiable properties.
- *Generic programming* (i.e., an OO programming technique enabling code to be written in terms of data types unknown until runtime [45]) was used to encode most of the functionality of manageability adaptors in a base abstract class, and to obtain resource-specific manageability adaptors by parameterising this abstract class with the dynamically generated resource data types.

Based on these requirements, J2EE and .NET were selected as candidate development environments for the prototype engine, with .NET being eventually preferred due to its better handling of dynamic proxy generation and slightly easier-to-use implementation of reflection.

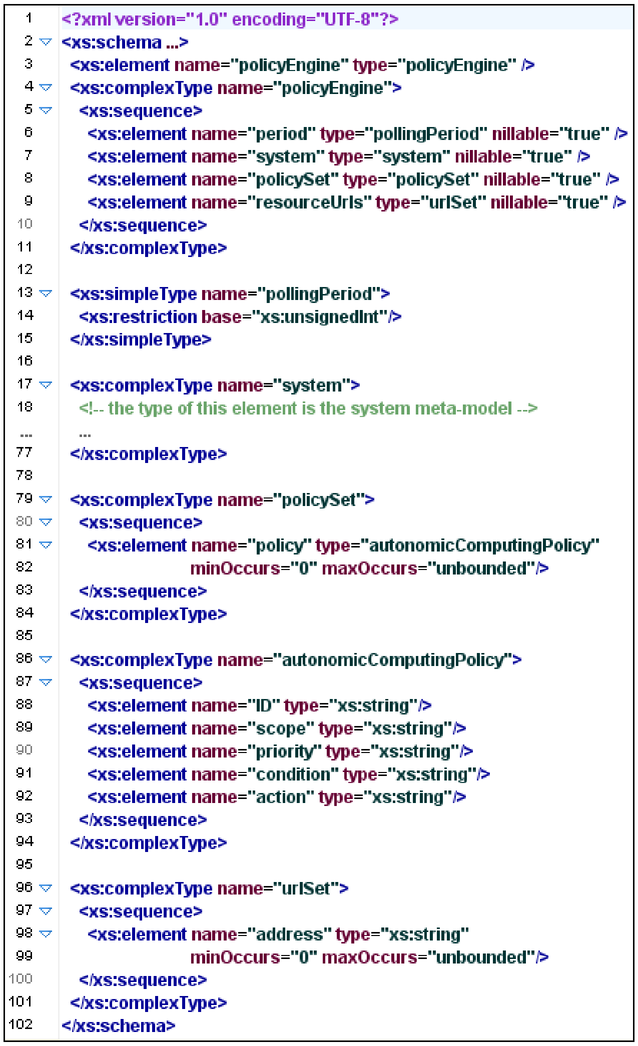
In order to ensure that one instance of the policy engine can be configured to manage other policy engine instances as required by our framework, we started by modelling the policy engine as an instance of the system meta-model in Figure 4. The resulting model (depicted in Figure 5) defines the properties (i.e., the parameters) of the policy engine, namely:

1. The policy evaluation period, in seconds (i.e., 'period').



```
<?xml version="1.0" encoding="UTF-8"?>
<system ...>
 <name>Universal Policy Engine</name>

 <resource>
  <ID>policy engine</ID>

  <!-- WHEN to manage: period -->
  <property>
   <ID>period</ID>
   <propertyDataType>
    <xs:element name="period" type="pollingPeriod"/>
    <xs:simpleType name="pollingPeriod">
     <xs:restriction base="xs:unsignedInt"/>
    </xs:simpleType>
   </propertyDataType>
   <mutability>mutable</mutability>
   <modifiability>read-write</modifiability>
   <subscribeability>false</subscribeability>
   <primaryKey>false</primaryKey>
  </property>

  <!-- WHAT to manage: managed system model -->
  <property>
   <ID>system</ID>
   <propertyDataType> [53 lines]
   <mutability>mutable</mutability>
   <modifiability>read-write</modifiability>
   <subscribeability>false</subscribeability>
   <primaryKey>false</primaryKey>
  </property>

  <!-- HOW to manage: policy set -->
  <property>
   <ID>policySet</ID>
   <propertyDataType>
    <xs:element name="policySet" type="policySet"/>
    <xs:complexType name="policySet">
     <xs:sequence>
      <xs:element name="policy" type="autonomicComputingPolicy"
                  minOccurs="0" maxOccurs="unbounded"/>
     </xs:sequence>
    </xs:complexType>
    <xs:complexType name="autonomicComputingPolicy">
     <xs:sequence>
      <xs:element name="ID" type="xs:string"/>
      <xs:element name="scope" type="xs:string"/>
      <xs:element name="priority" type="xs:string"/>
      <xs:element name="condition" type="xs:string"/>
      <xs:element name="action" type="xs:string"/>
     </xs:sequence>
    </xs:complexType>
   </propertyDataType>
   <mutability>mutable</mutability>
   <modifiability>read-write</modifiability>
   <subscribeability>false</subscribeability>
   <primaryKey>false</primaryKey>
  </property>

  <!-- WHERE to manage: managed resource address(es) -->
  <property>
   <ID>resourceUrls</ID>
   <propertyDataType> [7 lines]
   <mutability>mutable</mutability>
   <modifiability>read-write</modifiability>
   <subscribeability>false</subscribeability>
   <primaryKey>false</primaryKey>
  </property>
 </resource>
</system>
```

Figure 5. Policy engine model



```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema ...>
 <xs:element name="policyEngine" type="policyEngine" />
 <xs:complexType name="policyEngine">
  <xs:sequence>
   <xs:element name="period" type="pollingPeriod" nillable="true" />
   <xs:element name="system" type="system" nillable="true" />
   <xs:element name="policySet" type="policySet" nillable="true" />
   <xs:element name="resourceUrls" type="urlSet" nillable="true" />
  </xs:sequence>
 </xs:complexType>

 <xs:simpleType name="pollingPeriod">
  <xs:restriction base="xs:unsignedInt"/>
 </xs:simpleType>

 <xs:complexType name="system">
  <!-- the type of this element is the system meta-model -->
  ...
 </xs:complexType>

 <xs:complexType name="policySet">
  <xs:sequence>
   <xs:element name="policy" type="autonomicComputingPolicy"
               minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
 </xs:complexType>

 <xs:complexType name="autonomicComputingPolicy">
  <xs:sequence>
   <xs:element name="ID" type="xs:string"/>
   <xs:element name="scope" type="xs:string"/>
   <xs:element name="priority" type="xs:string"/>
   <xs:element name="condition" type="xs:string"/>
   <xs:element name="action" type="xs:string"/>
  </xs:sequence>
 </xs:complexType>

 <xs:complexType name="urlSet">
  <xs:sequence>
   <xs:element name="address" type="xs:string"
               minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
 </xs:complexType>
</xs:schema>
```

Figure 6. XML schema for a policy engine "resource"

2. The model of the managed system ('system'). Note that the 'propertyDataType' of this policy engine property (not shown in Figure 5 for the sake of conciseness) is the system meta-model from Figure 4, in its XML schema representation.

3. The set of policies to implement ('policySet'). Each such policy is an instance of a complex data type whose elements are described later in this section.

4. The locations of the resources to be managed ('resourceUrls'), which for the current version of the prototype are set explicitly (the use of a discovery technique [40] is intended for future versions).

A simple XSLT [46] (model) transformation that we implemented was used to generate the "policy engine" XML schema in Figure 6 from the policy engine model, then a **policyEngine** C# class was generated automatically from this schema using the off-the-shelf XML Schema Definition (XSD) tool [47] (Figure 7).

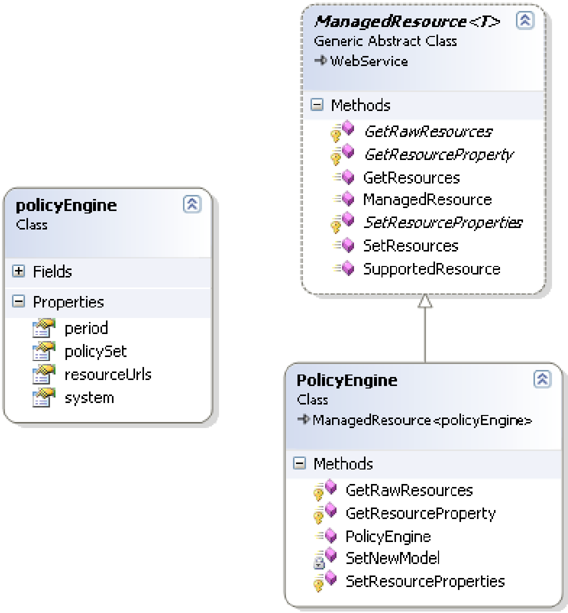


Figure 7. Policy engine resource (i.e., policyEngine) and manageability adaptor (i.e., PolicyEngine)—class diagram

Like for any other resource in our autonomic architecture from Figure 1, the parameters of the policy engine are accessed through a manageability adaptor. As shown in Figure 7, this adaptor (i.e., **PolicyEngine**) is a subclass of ***ManagedResource***$< T >$, the base class for all our manageability adaptors. The generic abstract class ***ManagedResource***$< T >$ comprises three web methods:

- SupportedResource returns the ID of the supported resource type.

- GetResources returns the list of all available resource instances. The method takes as argument a list of resource property IDs, and only the values of these properties are assessed and returned to the caller, thus preventing unnecessary resource property evaluation.

- SetResources takes as argument a list of resources of the supported type, and assigns any new values specified by the caller for the resource properties declared modifiable in the system model. The resources whose properties need to be modified are uniquely identified by the value of the resource properties marked as "primary key" components in the system model.

These web methods rely on resource-specific methods declared abstract in ***ManagedResource***$< T >$, and which any of its subclasses (including **PolicyEngine**) implements:

- *GetRawResources* builds a list of all available resource instances. The values of the resource properties need not be provided by this method.
- *GetResourceProperty* takes as arguments a resource instance and the ID of a resource property, and ensures that the property value is set in the resource object. The method is used by GetResources to fill in the required property values after obtaining a "raw" resource list from *GetRawResources*.
- *SetResourceProperties* takes a resource object and ensures that the modifiable properties of the corresponding real-world resource are assigned any new values specified in the resource object.

The web methods of our prototype, web service implementation of the policy engine correspond to the high-level sensors and effectors from the policy engine architecture in Figure 3. These methods can be used to read as well as to modify the engine parameters, ensuring that the parameters of the engine can be set by any type of software component that can be interfaced with a web service. For our case studies, we chose to implement a web-based administration tool that allows the remote configuration of the policy engine using a web browser (Figure 8), but this is by no means the only option available.

The four policy engine parameters that our administration tool reads and modifies using the web methods provided by the **PolicyEngine** manageability adaptor are described below. Note that these parameters are read by the tool whenever its front-end web page (shown in Figure 8) is loaded into a web browser, and modified when the administrator of the autonomic system uses the controls on this web page to explicitly operate a change in the engine parameters.

System model This parameter is an instance the XML system model described in Section 3.1. Changes to the 'system' property of the policy engine represent *reconfigurations of the engine for the management of new types of ICT resources*. This ability to specify the types of resources to be managed by the policy engine at runtime (and to change this specification as and when needed) represents a key feature of our autonomic computing architecture, and the reason why we term the policy engine a *reconfigurable* policy engine.[3] Internally, this operation involves:

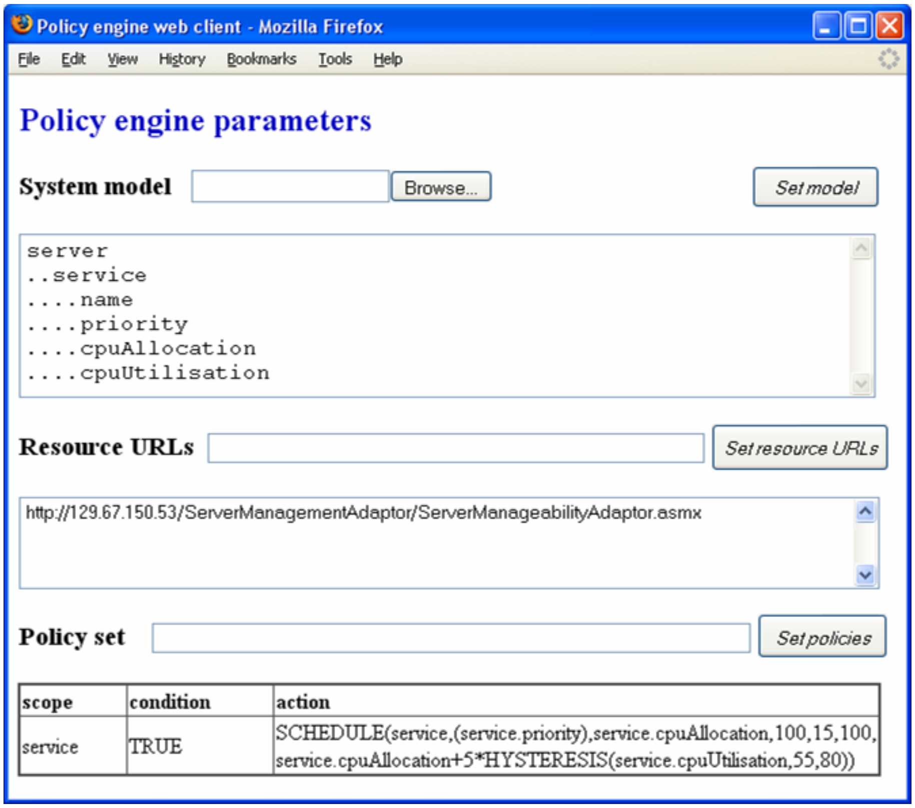
- The automated generation of data types (i.e., C# classes) for the new types of ICT resources. The steps involved in the generation of these classes are those described above for the policy engine itself: first, the XSLT (model) transformation mentioned earlier in this section is applied to the newly supplied system model and an XML schema for the new resource types is obtained; then, the XSD tool [47] is employed to generate the necessary classes.
- The automated generation of proxies for the manageability adaptor web services associated with the new resource types. In the .NET framework, this amounts to generating a Web Service Description Language (WSDL) file and two "discovery" files for each type of manageability adaptor, and deploying these files into a subdirectory of the policy engine. Templates for each of these files are kept within the policy engine. This enables the engine to generate the manageability adaptor WSDL file for a new resource type by simply replacing a couple of placeholders in its template WSDL file with the identifier and the XML-encoded type for the new resource, respectively—both fields being available from the system model. As concerns the two "discovery" files, these are identical copies of the templates maintained within the policy engine.

Resource URLs This parameter is a space-separated list of URLs, each of which represents the address of a manageability adaptor for a set of resources to be managed by the policy engine. Changes to the resource URLs trigger the engine to contact the manageability adaptors at the specified addresses in order to establish the type of resources they expose. If these manageability adaptors exist and they support an ICT resource type defined in the system model used to configure the policy engine, then the policy engine will take into account all resources exposed through these manageability adaptors when implementing the user-supplied policies. Manageability adaptors associated with resource types unknown to the engine are ignored until such time as a system model defining these resource types has been provided to the policy engine.

Policy set This parameter is a space-separated list of policies that the system administrator can type directly into the web-based administration tool in Figure 8. Each of these policies consists of three of the policy components described in Section 2.4, i.e., scope, condition and action. The fourth policy component (i.e., policy value) is not supported by the current version of the policy engine.

Policy changes lead to a re-analysis of the policy set and to its parsing into an internal format that makes the period-

[3] Clearly, other elements of an autonomic system implemented using our framework will undergo (re)configuration too, e.g., as a result of implementing the policies supplied to the policy engine. Such *self-configurations* are a defining characteristic of autonomic systems, and are discussed in detail elsewhere [5, 6, 7, 10, 12].



```
server
..service
....name
....priority
....cpuAllocation
....cpuUtilisation
```

http://129.67.150.53/ServerManagementAdaptor/ServerManageabilityAdaptor.asmx

scope	condition	action
service	TRUE	SCHEDULE(service,(service.priority),service.cpuAllocation,100,15,100, service.cpuAllocation+5*HYSTERESIS(service.cpuUtilisation,55,80))

Figure 8. Snapshot of the web client used to configure the policy engine

ical evaluation of policies computationally efficient. Only policies referring to known types of resources and manipulating their properties in valid ways (e.g., a policy must not attempt to modify a "read-only" resource property) are accepted.

Period This parameter is an integer numerical value that represents the policy evaluation period, expressed in seconds.

The operators that the current version of the policy engine supports within the policy scope, value, condition and action expressions (cf. Figure 2) are the operators for the manipulation of primitive data types and only a few of the more sophisticated operators recommended in Section 2.4 (e.g., set comprehension and scheduling). Support for additional operators is added on a regular basis as new case studies are being explored.

Also, the current version of the policy engine comprises only a subset of the policy engine components presented in Section 2.5, namely the components that are shaded in Figure 3. These components were selected so as to speed up the completion of a prototype that could be used to assess the effectiveness of the framework, and to explore the feasibility of our approach in an area in which no research has been conducted so far, namely the runtime, model-based reconfiguration of autonomic computing policy engines.

4. Case study

This section presents two autonomic solutions for the allocation of server capacity to a set of services, one employing action policies and taken from [1] and the other one using utility-function policies. This choice of a case study was motivated by the importance that this real-world application has had since the release of server-level capacity control APIs such as [37, 48]. Additionally, our prior experience with data-centre resource management [15] helped significantly during the implementation of the two solutions, and in the interpretation of the case study results.

Note that effective autonomic solutions for case studies from other application domains were also developed using our generic autonomic framework and its SOA implementation presented in this article, including dynamic power management, adaptive control of cluster availability within data-centres, and dynamic generation of web content. All of these case studies are described in detail in [49].

```
<system>
  <name>server</name>
  <resource>
    <ID>service</ID>
    <property>
      <ID>name</ID>
      <propertyDataType>
         <xs:simpleType name="serviceName">
           <xs:restriction base="xs:string"/>
         </xs:simpleType>
      </propertyDataType>
      <mutability>constant</mutability>
      <modifiability>read-only</modifiability>
      <subscribeability>false</subscribeability>
      <primaryKey>true</primaryKey>
    </property>
    <property>
      <ID>priority</ID>
      ...
    </property>
    <property>
      <ID>cpuAllocation</ID>
      <propertyDataType>
        <xs:simpleType name="serviceCpuAllocation">
          <xs:restriction base="xs:int">
            <xs:minExclusive value="0"/>
            <xs:maxInclusive value="100"/>
          </xs:restriction>
        </xs:simpleType>
      </propertyDataType>
      <mutability>mutable</mutability>
      <modifiability>read-write</modifiability>
      <subscribeability>false</subscribeability>
      <primaryKey>false</primaryKey>
    </property>
    <property>
      <ID>cpuUtilisation</ID>
      ...
    </property>
  </resource>
</system>
```

Figure 9. XML model of the managed system

4.1. Server capacity allocation using action policies

In order to test the SOA implementation of our autonomic computing framework, we configured a running instance of the policy engine from Section 3 to allocate the CPU capacity of a server to a set of services of different priority, and subjected to variable workloads. The only resource defined in the server model (Figure 9) was `service` with four properties: a unique `name`, an integer `priority`, the percentage of the server CPU allocated to the service (`cpuAllocation`) and the amount of CPU utilised by the service, expressed as a percentage of its CPU allocation (`cpuUtilisation`).

The policy depicted in Figure 8 allocates a percentage of the CPU capacity of the server to each 'service' resource, as selected by the policy scope. The 'TRUE' policy condition requires that the policy action is applied at all times (i.e., in line with the policy evaluation period of the engine). The policy action is specified by means of an expression that uses the SCHEDULE(R, $ordering$, $property$, $capacity$, min, max, $optimal$) operator that

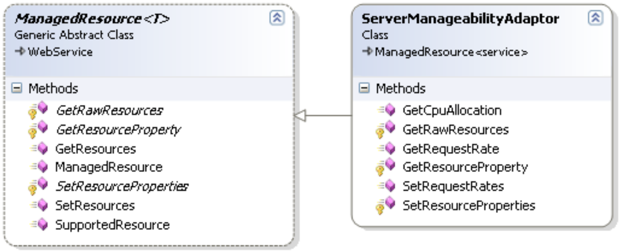


Figure 10. The server manageability adaptor

- sorts the resources in R in non-increasing order of the comparable expressions in $ordering$;
- in the sorted order, sets the specified resource $property$ to a value never smaller than min or larger than max, and as close to $optimal$ as possible;
- ensures that the overall sum of all $property$ values does not exceed the available $capacity$.

Accordingly, the policy action

$$\text{SCHEDULE}(service, \langle service.priority \rangle, service.cpuAllocation, 100, 15, 100, service.cpuAllocation + 5 * \text{HYSTERESIS}(service.cpuUtilisation, 55, 80))$$

in Figure 8 will set the $cpuAllocation$ property of all services to a value between 15% and 100%, subject to the overall CPU allocation staying within the 100% available capacity. Optimally, $cpuAllocation$ should be left unchanged if $55 \leq cpuUtilisation \leq 85$;[4] decreased by 5(%) if $cpuUtilisation < 55$;[5] and increased by 5(%) if $cpuUtilisation > 85$.[6] Note that this adjustment is performed repetitively, with a period given by the policy evaluation period parameter of the policy engine.

Like the policy engine itself, the manageability adaptor used to interface the engine with the server was implemented as a sub-class of ***ManagedResource***$< T >$—Figure 10.

The policy engine was then configured to manage remotely a server simulator running a high-priority 'premier' service and a lower-priority 'standard' service. The two services handled simulated user requests with exponentially-distributed inter-arrival time and normally-distributed processing time. Figure 11 shows the change in the system

[4] The HYSTERESIS($val, lower, upper$) operator used to achieve this behaviour returns -1, 0 or 1 if $val < lower$, $lower \leq val \leq upper$ or $upper < val$, respectively.

[5] The current CPU allocation is underutilised in this case, so it is decreased to avoid waste of CPU capacity.

[6] In this case, the service is utilising almost all CPU allocated to it, running the risk of becoming under-provisioned.

parameters when the request inter-arrival time of the two services was varied to simulate different workloads, and the policy engine was configured to implement the policy described earlier in this section; the system behaviour over the time intervals a to h is described below:

a. Both services are lightly loaded (5000μs request inter-arrival time) and have the minimum amount of CPU allocated (i.e., 15% each).

b. The load increases for the standard service, and its allocated CPU is increased by the policy engine accordingly.

c. For a brief period of time, the standard service uses its allocated CPU completely; no requests timeout though as its CPU allocation is increased swiftly.

d. The premium service workload starts to increase, and the policy engine increases its CPU allocation. Accordingly, the standard service starts to get less CPU.

e. As the workload for the premium service peaks and the policy engine schedules additional CPU capacity for this service, the standard service is allocated insufficient CPU and some of its client requests time out.[7]

f. The inter-arrival time for the premium service increases, and some of the CPU capacity allocated to it during the previous time interval is re-deployed by the policy engine to the standard service. No more requests time out.

g. Under constant workload, the CPU allocation is mostly stable.

h. To explore the role of the hysteresis, we replaced the hysteresis term in the policy action with HYSTERESIS$(service.cpuUtilisation, 80, 80)$, thus eliminating the hysteresis. This led to significant oscillations in the CPU capacity allocated to the services. The reinstatement of the original policy after this time interval brings the system back into a stable state.

The policy evaluation period was set to 3 seconds for this experiment, so that the system could self-adapt to the rapid variation in the workload of the two services. This allowed us to measure the CPU overhead of the policy engine, which was under 1% with the engine service running on a 1.8 GHz Windows XP machine. In a real scenario, such variations in the request inter-arrival time are likely to happen over longer intervals of time, and the system would successfully self-configure with far less frequent policy evaluations.

Note also that since the policy engine service is implemented as a managed resource, its policy evaluation period can be adjusted by another policy engine instance, so that it stays in step with the rate of change in the request inter-arrival time—a scenario that we are in the process of experimenting with.

4.2. Server capacity allocation using utility-function policies

We showed in the previous section that our framework can be used successfully to develop a realistic autonomic solution. However, the cost-effectiveness of this solution is limited by its usage of action policies designed by a system administrator with in-depth knowledge about the system resources. In this section, we describe how the same self-management capability can be realised by means of utility-function policies that can be designed by someone aware of the high-level business goals of the system but who has limited knowledge about its internal operation.

To implement utility-function policies, the policy engine needs an understanding of the *behaviour* of the system and its resources. Given a resource, we define its state $\mathbf{s}$ as the vector whose elements are the read-only properties of the resource, and its configuration $\mathbf{c}$ as the vector comprising its modifiable (i.e., read-write and write-only) properties. Let S and C be the value domains for $\mathbf{s}$ and $\mathbf{c}$, respectively.[8] A behavioural model of the resource is a function

$$behaviouralModel : S \times C \to S, \qquad (1)$$

such that for any current resource state $\mathbf{s} \in S$ and for any resource configuration $\mathbf{c} \in C$, $behaviouralModel(\mathbf{s}, \mathbf{c})$ represents the future state of the resource if its configuration is set to $\mathbf{c}$.

In practice, the policy engine works with an approximation of the behavioural model that consists of a set of discrete values of the $behaviouralModel$ in (1)—an approach that works well with the continuous behavioural models that are typical to most real-world systems. As a further simplification, any state and configuration components that play no role in the resource behaviour (e.g., the `name` and `priority` properties of the `service` resource in our system) are disregarded in the behavioural model approximation that the policy engine operates with.

There are multiple ways in which the policy engine can acquire the behavioural model required to support utility-function policies. The two extreme ones are to have this model supplied by the resource itself and to have the model generated automatically by the machine learning modules within the policy engine (see Figure 3). An intermediate option is to have an initial behavioural model supplied to the policy engine, and further refined by its machine learning modules. Our prototype policy engine does not include the machine learning modules, hence the required

[7] Requests time out after spending T=5s in a service request queue.

[8] Note that S and C are fully specified in the system model.

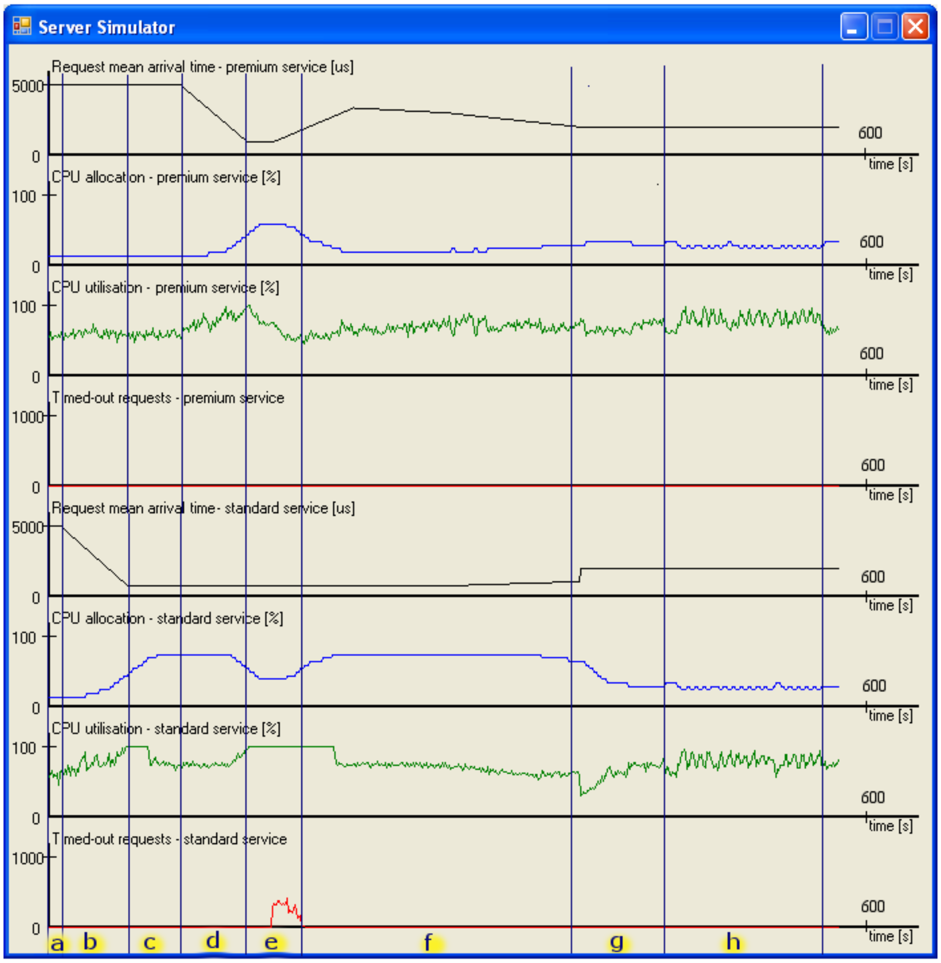

Server Simulator
Request mean arrival time - premium service [us]
5000
0
600
time [s]
CPU allocation - premium service [%]
100
0
600
time [s]
CPU utilisation - premium service [%]
100
0
600
time [s]
Timed-out requests - premium service
1000
0
600
time [s]
Request mean arrival time - standard service [us]
5000
0
600
time [s]
CPU allocation - standard service [%]
100
0
600
time [s]
CPU utilisation - standard service [%]
100
0
600
time [s]
Timed-out requests - standard service
1000
0
600
time [s]
a
b
c
d
e
f
g
h

Figure 11. Snapshot of a typical server simulation experiment

behavioural model is provided by the manageability adaptor for the `service` resource. This behavioural model (Figure 12) describes how the response time of a service varies with the request inter-arrival time and the percentage of server CPU allocated to the service, and was obtained from multiple runs of the server simulator in which the average service response time was recorded for 920 equidistant points covering the entire (`interArrivalTime`, `cpuAllocation`) value domain.

To use utility-function policies in our autonomic solution, we added several new `service` properties to the system model devised in the previous section (Figure 13):

- `responseTime`, the service response time, measured in milliseconds and averaged over the past one-second time interval;
- `interArrivalTime`, the mean request inter-arrival time;
- `behaviouralModel`, an approximation of the service behavioural model.

We then defined a utility function that models the business gain associated with running $n > 0$ services with different levels of service:

$$utility(R) = \sum_{r\in R} r.priority * \min(1000,$$
$$\max(0, 2000 - r.responseTime)), \qquad (2)$$

where R is the set of `service` resources. Figure 14 depicts the utility function for a server running a “premium” service with priority 100 and a “standard” service with priority 10.

The policy implemented by the autonomic system was defined by means of the MAXIMIZE(R, $utility$, $property$, $capacity$, min, max, $model$) operator that uses the information about the system behaviour encoded in $model$ to set the value of the specified resource $property$ for all resources in R such as to:

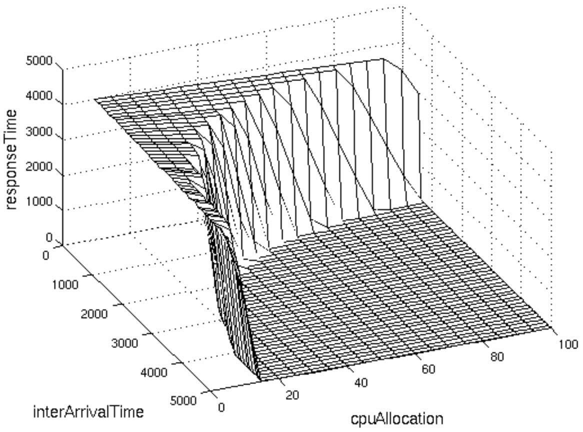


Figure 12. Service behavioural model

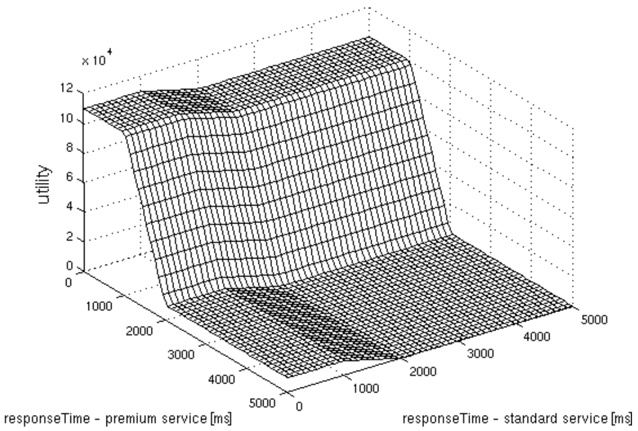


Figure 14. Utility function

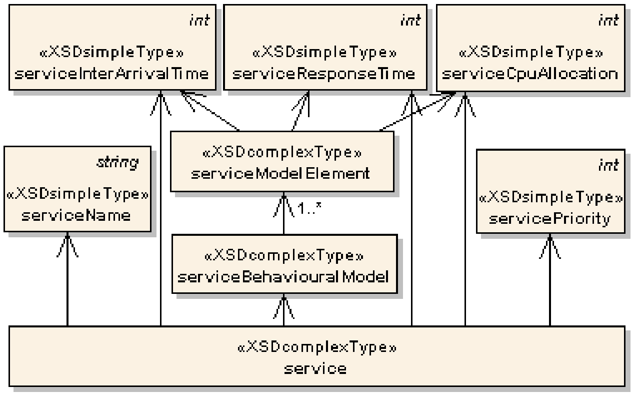


Figure 13. Service model for Section 4.2

- maximize the value of the $utility$ function;
- ensure that the value of $property$ stays between min and max, and that the sum of the $property$ values across all resources in R does not exceed the available $capacity$.

The arguments of MAXIMIZE were specified as shown in Table 1, in order to supply the policy engine with the definition of the utility function, and to link the `responseTime`, `interArrivalTime` and `cpuAllocation` properties of a `service` resource to the components of its `behaviouralModel` property. Each time it evaluates the utility-function policy, the policy engine uses this information to select the elements from the behavioural model that are in the proximity of the current state of the system; the Euclidean metric is used for this calculation. The new configuration for the system is then chosen as the one associated with the selected element that maximizes the value of the utility function.

Note that the policy engine could be required to synthesise the behavioural model itself by specifying the $model$ argument of MAXIMIZE as "$service.responseTime(service.interArrivalTime,$ $service.cpuAllocation)$", so as to indicate only that the service response time depends on the request inter-arrival time and the CPU allocation for the service. This syntax will be used when machine learning support is added to the policy engine prototype.

Figure 15 illustrates a typical experiment in which the utility-function policy described in this section was used to manage the allocation of CPU to the same two services as in Section 4.1. The experimental results resemble those obtained when an action policy was used (Figure 11), therefore confirming the effectiveness of our approach to developing autonomic solutions that use utility-function policies in conjunction with a behavioural model of the managed resources. The few differences between the two sets of experimental results indicate that the autonomic solution that uses utility-function policies is actually superior to the solution based on action policies, as shown by these differences across the time intervals a to e in Figure 15:

a. Shortly after the utility-function policy is supplied to the policy engine, the CPU allocation is decreased to the minimum level that can ensure the optimal level of service. When the action policy was used, CPU variations of such magnitude required multiple policy evaluations.

b. The CPU allocated to the standard service increases in line with its workload.

c. The CPU allocation for the premium service also increases, but the response time of both services can still be maintained at values that maximize the utility function.

Table 1. Arguments of the MAXIMIZE **operator for Section 4.2.**

argument	value
R	$service$
$utility$	$\mathrm{SUM}(service.priority * \mathrm{MIN}(1000, \mathrm{MAX}(0, 2000 - service.responseTime)))$
$property$	$service.cpuAllocation$
$capacity$	100
min	15
max	100
$model$	$service.responseTime(service.interArrivalTime, service.cpuAllocation) =$ $service.behaviouralModel.responseTime($ $service.behaviouralModel.interArrivalTime, service.behaviouralModel.cpuAllocation)$

d. The amount of CPU required to satisfy the increased demand for the premium service leaves insufficient CPU capacity for the standard service to make any contribution to the utility function, hence it is allocated the minimum amount of CPU (15%). When the action policy was used, all CPU capacity not given to the premium service was allocated to the standard service even if the standard service was of no use to the business. In contrast, the utility-function policy allocates additional CPU to the standard service only when enough capacity is available to bring this service into a region of operation in which it can contribute to the utility function.

e. The response time for the standard service is recovering slowly, as it takes time to drain the request queue built during the previous time interval. The use of an enhanced behavioural model that takes into account the length of the service request queue should speed up this recovery.

f. The CPU allocations for the two services are constant over long periods of time. With action policies, this could be achieved only by explicitly including a hysteresis construct in the policy specification.

Note that in order to outperform solutions based on action policies (as demonstrated by our case study), utility-function policies need to employ "adequately specified" utility functions. From our experience with developing policy-based autonomic solutions for data-centre resource management, devising effective utility functions for medium-sized applications requires in-depth knowledge of the application domain and careful validation before deployment within a production system, but is a task that can be completed successfully by an experienced system administrator. When optimal utility functions are sought, multiple (and possibly conflicting) system objectives need to be captured by these functions and/or large-scale, complex systems are involved in the intended autonomic applications, devising the utility functions is much more difficult. The development of techniques for the construction of such utility functions represents an active research area in autonomic computing.

5. Related work

The autonomic infrastructure proposed in [50] is retrofitting autonomic functionality onto legacy systems by using *sensors* to collect resource data, *gauges* to interpret these data and *controllers* to decide the "adaptations" to be enforced on the managed systems through *effectors*. This infrastructure was successfully used to monitor, analyse and control legacy systems in applications such as spam detection, instant messaging quality-of-service management and load balancing for geographical information systems [51]. Our generic autonomic framework addresses several key areas that are not supported by the approach in [50, 51]. By using a system model for the configuration of its policy engine, our architecture can be used for the autonomic management of heterogeneous types of resources. Moreover, our managed system can include resources beyond the software components handled by the infrastructure in [50].

In [52], the authors define an autonomic architecture meta-model that extends IBM's autonomic computing blueprint [13], and use a model-driven process to partly automate the generation of instances of this meta-model. Each instance is a special-purpose *organic computing system* that can handle the use cases defined by the model used for its generation. Our general-purpose autonomic architecture eliminates the need for the 19-step generation process described in [52] by using a policy engine that can be dynamically reconfigured to handle any use cases encoded within its system model and policy set.

A number of other projects have investigated isolated aspects related to the development of autonomic systems out of non-autonomic components. Some of these projects addressed the standardisation of the policy information model,

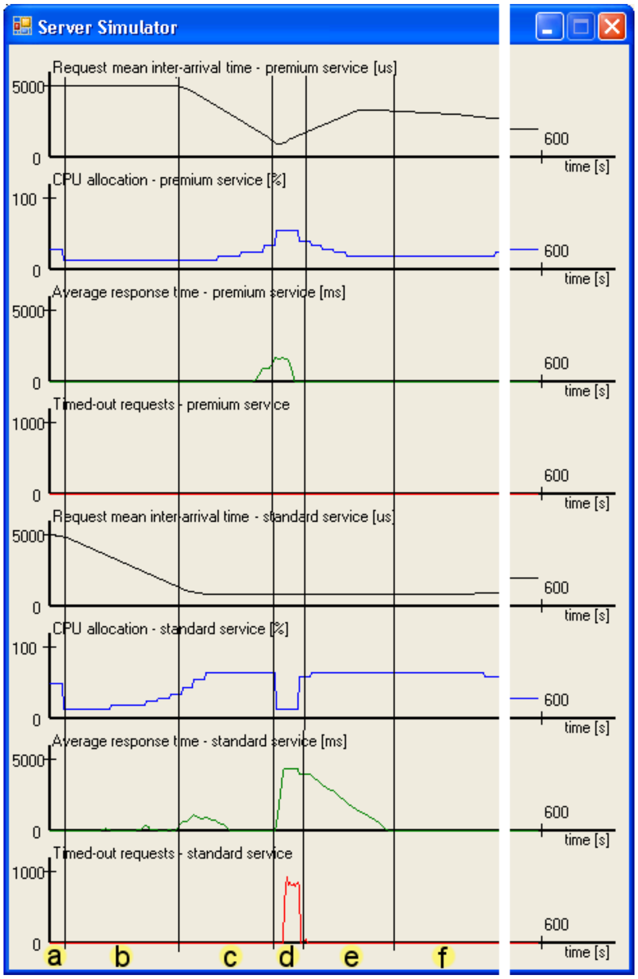

Server Simulator
Request mean inter-arrival time - premium service [us]
5000
0
CPU allocation - premium service [%]
100
0
Average response time - premium service [ms]
5000
0
Timed-out requests - premium service
1000
0
Request mean inter-arrival time - standard service [us]
5000
0
CPU allocation - standard service [%]
100
0
Average response time - standard service [ms]
5000
0
Timed-out requests - standard service
1000
0
a
b
c
d
e
f
600
time [s]

Figure 15. Utility-function results

with the Policy Core Information Model [34] representing the most prominent outcome of this work. Recent efforts such as Oasis' Web Services Distributed Management (WSDM) project were directed at the standardisation of the interfaces through which the manageability of a resource is made available to other applications [25]. An integrated development environment for the implementation of WSDM-compliant interfaces is currently available from IBM [26].

In a different area, expression languages were proposed for the specification of policy conditions and actions, and used to implement a range of policies [30, 33, 53, 54]. In addition to the development of standards and technologies, complete autonomic computing solutions have been produced recently [15, 37, 48], typically for the management of specific systems, and with limited ability to function in different scenarios from those they were originally intended for.

6. Conclusion

We described the SOA-based implementation of a generic framework intended to simplify significantly the development of autonomic systems, and thus to establish autonomic computing as a cost-effective approach to handling the spiralling complexity of today's computer systems. The ability to dynamically reconfigure the policy engine employed by the framework ensures that it can be used to build self-managing systems out of legacy and autonomic-enabled ICT resources whose characteristics are unknown until runtime, all without any modification to these resources or the policy engine.

Experimental work was carried out to validate the effectiveness of the SOA implementation of our autonomic computing framework. In this article, we presented a case study involving the development of two autonomic solutions for the allocation of server capacity to services of different priorities and varying workloads. The experimental results showed that our general-purpose framework could perform the planned management task successfully, and similarly to a dedicated commercial system for data-centre resource management [15, 55]. However, unlike the commercial resource-management system, our novel approach has the unique ability to handle resources whose types are unknown at implementation and deployment time, therefore enabling the cost-effective development of autonomic solutions across a broad variety of application domains—additional case studies from other application domains are described in [49].

The experimental results from the case studies presented in this article and in [49] suggest that the overheads associated with the evaluation of autonomic computing policies for realistic applications involving small ICT systems are acceptable. Thus, the realisation of relatively sophisticated utility-function policies far outweighs the observed utilisation of 1-2% of the CPU capacity and of a negligible amount of the memory of a low-end server. Furthermore, notice that this server need not be part of the managed system if self-management capabilites are added to a production ICT system that might be sensitive to such overheads: given its implementation as a web service, the policy engine can be deployed on a dedicated server. The only components to be deployed on the production system when this approach is used are the low-footprint manageability adaptors.

Clearly, the overhead levels mentioned above are characteristic of applications involving small to medium-sized ICT systems similar to those considered in our cases studies. Further work is planned to assess the scalability of the framework to large and very large ICT systems like those encountered in today's data centres.

Ongoing work is also dedicated to augmenting the SOA implementation by adding the policy engine components and functionality specified by our framework but which are not supported by its current version (cf. Figure 3). In particular, we are looking at ways to integrate machine learning [41] into the prototype, along the lines of the work described in [56, 57]. Another research topic requiring investigation is the automated synthesis of effective autonomic computing policies, for instance for the scenario in which one instance of the policy engine is tasked with managing another instance of the same architecture, as described in the article.

Finally, devising "good" utility-function policies for complex ICT systems and avoiding conflicts within sets of such policies represent open research questions for the autonomic computing community. It is hoped that the availability of generic development frameworks such as the one described in this article will help address these questions by re-directing much of the effort involved in developing an autonomic system away from the implementation of its components and towards the design and analysis of its autonomic computing policies.

Acknowledgement This work was partly supported by the UK Engineering and Physical Sciences Research Council grant EP/F001096/1.

References

[1] R. Calinescu, "Implementation of a generic autonomic framework," in *Fourth International Conference on Autonomic and Autonomous Systems (ICAS 2008)*, D. Greenwood M. Grottke, H. Lutfiyya and M. Popescu, Eds., March 2008, pp. 124–129.

[2] T. Lenard and D. Britton, *The Digital Economy Factbook*. The Progress and Freedom Foundation, 2006.

[3] D. Tapscott, *Digital Economy: Promise and Peril in the Age of Networked Intelligence*. McGraw-Hill, 1997.

[4] Y. Bar-Yam, M. A. Allison, R. Batdorf, H. Chen, H. Generazio, H. Singh and S. Tucker, "The characteristics and emerging behaviors of system-of-systems," New England Complex Systems Institute, Tech. Rep., January 2004.

[5] IBM Corporation, "Autonomic computing: IBM's perspective on the state of information technology," October 2001.

[6] R. Murch, *Autonomic Computing*. IBM Press, 2004.

[7] J. O. Kephart and D. M. Chess, "The vision of autonomic computing," *IEEE Computer Journal*, vol. 36, no. 1, pp. 41–50, January 2003.

[8] S. Dobson, S. Denazis, A. Fernndez, D. Gaiti, E. Gelenbe, F. Massacci, P. Nixon, F. Saffre, N. Schmidt and F. Zambonelli, "A survey of autonomic communications," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 1, no. 2, pp. 223–259, December 2006.

[9] M. Hinchey and R. Sterritt, "Self-managing software," *Computer*, vol. 39, no. 2, pp. 107–109, February 2006.

[10] M. Parashar and S. Hariri, *Autonomic Computing: Concepts, Infrastructure & Applications*. CRC Press, 2006.

[11] R. Calinescu, "Challenges and best practices in policy-based autonomic architectures," in *Proc. 3rd IEEE Intl. Symp. Dependable, Autonomic and Secure Computing*, 2007, pp. 65–74.

[12] M. C. Huebscher and J. A. McCann, "A survey of autonomic computing—degrees, models, and applications," *ACM Comput. Surv.*, vol. 40, no. 3, pp. 1–28, 2008.

[13] IBM Corporation, "An architectural blueprint for autonomic computing," 2004, http://www-03.ibm.com/autonomic/pdfs/ACBP2_2004-10-04.pdf.

[14] R. Sterritt and M. Hinchey, "Biologically-inspired concepts for self-management of complexity," in *Proc. 11th IEEE Intl. Conf. Engineering of Complex Computer Systems*, 2006, pp. 163–168.

[15] R. Calinescu and J. Hill, "System providing methodology for policy-based resource allocation," July 2005, US Patent Application 20050149940.

[16] M. Devarakonda, D. Chess, I. Whalley, A. Segal, P. Goyal, A. Sachedina, K. Romanufa, E. Lassettre, W. Tetzlaff and B. Arnold, "Policy-based autonomic storage allocation," in *Self-Managing Distributed Systems*, ser. LNCS, vol. 2867. Springer, 2004, pp. 143–154.

[17] S. Ghanbari, G. Soundararajan, J. Chen and C. Amza, "Adaptive learning of metric correlations for temperature-aware database provisioning," in *Proc. 4th IEEE Intl. Conf. Autonomic Computing*, June 2007.

[18] C. Lefurgy, X. Wang and M. Ware, "Server-level power control," in *Proc. 4th IEEE Intl. Conf. Autonomic Computing*, June 2007.

[19] W.-S. Li, D. C. Zilio, V. S. Batra, M. Subramanian, C. Zuzarte and I. Narang, "Load balancing for multi-tiered database systems through autonomic placement

of materialized views," in *Proc. 22nd IEEE Intl. Conf. Data Engineering*, April 2006.

[20] R. Sterritt, M. Hinchey, C. Rouff, J. Rash and W. Truszkowski, "Sustainable and autonomic space exploration missions," in *Proc. 2nd IEEE Intl. Conf. Space Mission Challenges for Information Technology*, 2006, pp. 59–66.

[21] R. Calinescu, "Model-driven autonomic architecture," in *Proc. 4th IEEE Intl. Conf. Autonomic Computing*, 2007.

[22] R. Calinescu, "Towards a generic autonomic architecture for legacy resource management," in *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*, K. Elleithy, Ed. Springer, 2008, pp. 410–415.

[23] W. Walsh, G. Tesauro, J. O. Kephart and R. Das, "Utility functions in autonomic systems," in *Proc. 1st Intl. Conf. Autonomic Computing*, 2004, pp. 70–77.

[24] O. Zimmermann, M. Tomlinson and S. Peuser, *Perspectives on Web Services: Applying SOAP, WSDL and UDDI to Real-World Projects*. Springer, 2005.

[25] B. Murray, K. Wilson and M. Ellison, "Web Services Distributed Management: MUWS primer," February 2006, oASIS WSDM Committee Draft, http://www.oasis-open.org/committees/download.php/17000/wsdm-1.0-muws-primer-cd-01.doc.

[26] IBM Corporation, "Autonomic integrated development environment," April 2006, http://www.alphaworks.ibm.com/ tech/aide.

[27] Microsoft Corporation, "System Definition Model overview," April 2004, http://download.microsoft.com/download/b/3/8/b38239c7-2766-4632-9b13-33cf08fad522/sdmwp.doc.

[28] Microsoft Corporation, "Microsoft Dynamic Systems Initiative Overview," March 2005, http://download.microsoft.com/download/8/7/8/8783b65e-d619-46d7-aa8d-b4f13a97eeb0/DSIoverview.doc.

[29] J. Arwe, J. Boucher, P. Dublish, Z. Eckert, D. Ehnebuske, J. Hass, S. Jerman *et al.*, "Service Modeling Language, version 1.0," March 2007, http://www.w3.org/ Submission/2007/SUBM-sml-20070321.

[30] IBM Corporation, "Policy Management for Autonomic Computing, version 1.2," 2005, http://dl.alphaworks.ibm.com/technologies/pmac/PMAC12_sdd.pdf.

[31] L. Stojanovic, J. Schneider, A. Maedche, S. Libischer, R. Studer, T. Lumpp, A. Abecker, G. Breiter, and J. Dinger, "The role of ontologies in autonomic computing systems," *IBM Systems Journal*, vol. 43, no. 3, pp. 598–616, 2004.

[32] K. Breitman and M. Perazolo, "Using formal ontology representation and alignment strategies to enhance resource integration in multi vendor autonomic environments," in *Proceedings of the 4th IEEE International Workshop on Engineering of Autonomic and Autonomous Systems*, Tucson, AZ USA, 2007, pp. 117–126.

[33] D. Agrawal, J. Giles, K.-W. Lee and J. Lobo, "Autonomic Computing Expression Language (ACEL) 1.2: User's Guide," 2005, http://www-128.ibm.com/developerworks/edu/ac-dw-ac-acel-i.html.

[34] B. Moore, "Policy Core Information Model (PCIM) extensions," January 2003, iETF RFC 3460, http://www.ietf.org/rfc/rfc3460.txt.

[35] J. O. Kephart and W. E. Walsh, "An artificial intelligence perspective on autonomic computing policies," in *Proc. 5th IEEE Intl. Workshop on Policies for Distributed Systems and Networks*, 2004.

[36] S. White, J. E. Hanson, I. Whalley, D. M. Chess, and J. O. Kephart, "An architectural approach to autonomic computing," in *Proc. 1st IEEE Intl. Conf. Autonomic Computing*. IEEE Computer Society, 2004, pp. 2–9.

[37] Microsoft Corporation, "Windows System Resource Manager (WSRM) White Paper," August 2003, http://download.microsoft.com/download/a/7/a/a7a06462-1d80-4386-9505-91cca1e61940/WSRM%20Command-Line%20Interface.doc.

[38] J. Woodcock and J. Davies, *Using Z. Specification, Refinement and Proof*. Prentice Hall, 1996.

[39] M. Kwiatkowska, "Quantitative verification: Models, techniques and tools," in *Proc. 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering*. ACM Press, September 2007, pp. 449–458.

[40] R. Harbird, S. Hailes and C. Mascolo, "Adaptive resource discovery for ubiquitous computing," in *Proc. 2nd Workshop Middleware for Pervasive and Ad-hoc Computing*, ser. ACM Intl. Conference Proceeding Series, vol. 77, October 2004, pp. 155–160.

[41] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2007.

[42] R. Calinescu and M. Kwiatkowska, "Using quantitative analysis to implement autonomic IT systems," in *Proceedings of the 31st International Conference on Software Engineering (ICSE 2009)*, May 2009, to appear.

[43] OASIS, "Web Services Resource Metadata 1.0," November 2006.

[44] J. M. Sobel and D. P. Friedman, "An introduction to reflection-oriented programming," in *In Proceedings of Reflection96*, 1996.

[45] R. Garcia, J. Jarvi, A. Lumsdaine, J. G. Siek and J. Willcock, "A comparative study of language support for generic programming," *ACM SIGPLAN Notices*, vol. 38, no. 11, pp. 115–134, November 2003.

[46] "SAXON – The XSLT and XQuery Processor," http://saxon.sourceforge.net/.

[47] Microsoft Corporation, "Xml schema definition tool (xsd.exe)," 2007, http://msdn2.microsoft.com/en-us/library/x6c1kb0s(VS.80).aspx.

[48] Sun Microsystems, Inc, "Sun^{TM} Grid Compute Utility—Reference guide," June 2006, http://www.sun.com/service/sungrid/SunGridUG.pdf.

[49] R. Calinescu, "General-purpose autonomic computing," in *Autonomic Computing and Networking*, M. Denko *et al.*, Eds. Springer, June 2009.

[50] J. Parekh, G. Kaiser, P. Gross and G. Valetto, "Retrofitting autonomic capabilities onto legacy systems," *Cluster Computing*, vol. 9, no. 2, pp. 141–159, April 2006.

[51] G. Kaiser, J. Parekh, P. Gross and G. Valetto, "Kinesthetics extreme: An external infrastructure for monitoring distributed legacy systems," in *Proc. of the 5th Annual Intl. Active Middleware Workshop*, June 2003.

[52] H. Kasinger and B. Bauer, "Towards a model-driven software engineering methodology for organic computing systems," in *Proc. 4th Intl. Conf. Computational Intelligence*, July 2005, pp. 141–146.

[53] R. Anthony, "A policy-definition language and prototype implementation library for policy-based autonomic systems," in *Proceedings of the 4th IEEE International Conference on Autonomic Computing*, Dublin, Ireland, June 2006, pp. 265–276.

[54] N. Damianou, N. Dulay , E. Lupu and M. Sloman, "The Ponder policy specification language," in *Policies for Distributed Systems and Networks*, ser. LNCS, vol. 1995, Bristol, UK, 2001, pp. 18–38.

[55] B. McColl, "Intelligent, policy-driven orchestration of sensors and effectors across the data center in real-time," Sychron Inc, White paper, April 2004, http://hosteddocs.ittoolbox.com/BM042304.pdf.

[56] G. Tesauro, "Reinforcement learning in autonomic computing: A manifesto and case studies," *IEEE Internet Computing*, vol. 11, no. 1, pp. 22–30, January 2007.

[57] T. Lau, D. Oblinger, L. Bergman, V. Castelli and C. Anderson, "Learning procedures for autonomic computing," in *Proc. Workshop on AI and Autonomic Computing: Developing a Research Agenda for Self-Managing Computer Systems*, August 2003, http://tlau.org/research/papers/autonomic-ijcai2003.pdf.

[58] R. Calinescu, "Resource-definition policies for autonomic computing," in *Fifth International Conference on Autonomic and Autonomous Systems (ICAS 2009)*, IEEEComputer Society Press, April 2009, pp. 111–116.

Autonomic Service Control In Next Generation Networks

Michael Kleis[#], Andreas Klenk[*], Benoit Radier°, Sanaa Elmoumouhi°, Georg Carle[*],
and Mikael Salaun°

[*] *Fraunhofer FOKUS,*
Kaiserin-Augusta-Alle 31
10589 Berlin, Germany
michael.kleis@fokus.fraunhofer.de

[#] *Technische Universität München*
Boltzmannstraße 3
85748,Garching bei München, Germany
{klenk,carle}@net.in.tum.de

° *France Télécom R&D*
Avenue Pierre Marzin 2
22307 Lannion, France
{benoit.radier, sanaa.elmoumouhi, mikael.salaun}@orange-ft.com

Abstract

Current standardization efforts aim towards a unifying platform for fixed and mobile telecommunication services. The IP multimedia subsystem is advocated as the candidate for building Next Generation Networks (NGNs). However, the direction taken in standardization is towards a rather static architecture with centralized features. The downside is an expected increase in service management complexity and the need for highly specialized infrastructures. This paper presents an approach for improving service quality, scalability and reliability while facilitating service management towards self-managing Next Generation Networks. To approach this we utilize and combine functionality available in the network using a Peer-to-Peer based service composition mechanism. The construction of composed services is based on a service chain principle and incorporates information about available services, Quality of Service (QoS) and applicable Servie Level Agreements (SLAs).

1. Introduction

Fixed mobile convergence is a hot topic in telecommunications industry. An important building block for next generation converged networks is the IP Multimedia Subsystem (IMS) defined by 3GPP[1] and taken into account by TISPAN[2]. The IMS allows for different types of access technologies while allowing mobile usage as well as an easy service integration. The main approach in IMS standardization is to define functional components and interfaces. The technical realization of this architectural model is inherently centralized and usually demands for a careful administration and deployment. Even in the case that IMS components are very reliable, the failure of an IMS component can lead to service interruption. This fact combined with the increasing complexity of service provisioning can result in a high management and configuration overhead for future IMS based services and thus high costs. In addition it can be expected that the resources of IMS components have to be allocated for peak usage, and will most of the time be underutilized. Thus CAPEX and OPEX for new services can be high and as a consequence the IMS architecture may be in fact not as flexible as expected.

In contrast, the prospects of autonomic networking research are to allow the network to take care of itself and to resolve problems automatically. In fact, the success of Peer-to-Peer (P2P) technology for Voice-over-IP (e.g. Skype) has already proven the value of distributed self-organizing architectures for telephony. Thus the question arises: *If and how P2P and overlay*

[1] 3GPP: Third Generation Partnership Project

[2] TISPAN: Telecoms & Internet converged Services & Protocols for Advanced Networks

technology can be adopted for service platforms in NGNs?

In this paper we describe the approach followed by the research project *Situated Autonomous Service Control* (SASCO) to explore and develop a secure, overlay based platform for an autonomous service provisioning in NGNs. To address the above-named question we start with the premises from the viewpoint of a multimedia service provider. The core requirements of a solution covering multimedia processing as well as QoS aware transport and routing are low costs, low management and configuration complexity as well as scalability. Based on these requirements the core research challenge is the exploration of a self-* [10][12] system for service provisioning in future networks. In this paper self-* denotes self-configuring, self-organizing, self-managing and self-repairing. Extending the results published in [1] we concentrate on an overall picture containing the required core concepts. For a detailed discussion of technical aspects we refer to [13][14]. As one result, the aspired approach would change the way how subscriber, network operator and service provider interact in a beneficial way for all parties. In the past, two traditional business relationships with regard to service provisioning dominated:

1. A direct business relation between clients and service providers in networks based on the end-2-end principle [11] as e.g. the Internet. The network provider is offering essentially the same interface to the transport service to client and service provider.

2. A business relation between the client and an operator or the network provider as in networks based on the intelligent network principle. In such networks, new services have to be introduced either by the network provider itself or by a third party provider using a special interface (e.g. Parlay X defined by ETSI) offered by the network provider for this purpose.

We propose to combine the strengths of both principles with the aim of defining an architecture that can be the basis for future and autonomic networks. The resulting entity model is depicted in Figure 1. As a consequence the approach will allow:

- The service provider to concentrate on service/content provisioning and to abstract from transport or end user terminal related issues.
- The network provider to offer value added transport services as: media adaptation to client terminals and access technology, broadcast/multicast services, caching, as well as seamless services and connectivity for clients.

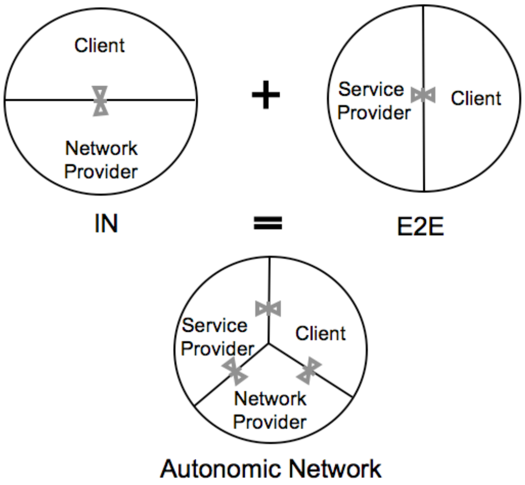


Figure 1 Entity Model

One of the main anticipated research challenges to realize this vision in the area of service composition is to resolve concrete service chains with a scalable distributed algorithm and to obey quality of service constraints imposed by the corresponding data transport and the services itself. In addition there is a need for explicit knowledge about the service chain to help the signalling between the partaking processing nodes and access control functions.

The paper is organized as follows: In Section 2 we present related work on autonomic overlay technology. Section 3 motivates the idea of autonomic service control. The following sections describe the decomposition and creation of service chains (Section 4), the DHT based control (Section 5), cooperative service provisioning (Section 6) and the integration of access control functionality (Section 7). We show how our work integrates withy existing IMS components in Section 8. We conclude in Section 9.

2. Related Work

In recent years one can observe a rising interest in overlay-related research. Since 2000, many classical network problems like QoS [24], Resilience [25], Multicast [26] or Security [23] have been addressed using the overlay approach. In addition, one branch of overlay-related research started to study the possibility of using an overlay concept for the flexible, on-demand composition of services.

One of the first projects in this direction was the Ninja Project [27]. The architecture developed by the project includes the notion of (logical and physical) service paths which have counterparts in many of the following proposals towards a overlay-based Service Composition as e.g. [28][29][30][31]. Further projects addressing service composition include SAHARA [34], SWORD [35] and the Ambient Networking Project [36]. The SAHARA project addressed trust and performance related aspects of service composition in case the component services are hosted by different providers. SWORD focused on the generation of service composition plans based on the requirements of the composed service with a strong focus on web service composition, while in Ambient Networks service composition was used for an on-demand processing and routing of media flows. Inside the overlay community the MONET group of Klara Nahrstedt at the University of Illinois Urbana-Campaign had a strong focus on network and QoS related questions [29][37][38][33][39][28][30][31]. Further contributions to this field can be found in [40][41] and [13]. In addition, from the perspective of active or programmable networks, routing problems closely related to the ones in overlay-based Service Composition have been addressed for example in [42] or [32]. Load balancing and stability issues for Service Composition have been discussed e.g. in [43].

In general, the overlay centric work towards a network and QoS aware composition of services can be classified into two main categories: 1. Centralized systems or systems which require global knowledge 2. Decentralized systems

Examples falling into the first category are [29][44][38][30][31][42][32]. As part of this category we consider approaches relying on a central point where service and QoS-related data is aggregated or schemes applying link state routing to address the Service Composition problem. In general, all schemes in this category require or assume permanent QoS measurements between all potential overlay nodes which results in a measurement overhead of $O(n^2)$ for n nodes. In addition it is necessary to either broadcast the measurement results and a description of offered services to the whole group, or to deliver them periodically to a central entity responsible for overlay setup.

This extensive measurement and dissemination overhead is required since in an overlay context the situation is different to the case of network layer link state protocols as e.g. OSPF. The reason for this is the fact that the nodes involved in overlays are usually end systems. For a given end system every other end system in the network is a potential overlay neighbor. Thus there is no notion of quasi-static network topology as in the case of layer three networking. Instead a Service Overlay topology is built up on demand and a link in the overlay in general corresponds to a path in the underlying network. As a consequence schemes in this category usually come with a large measurement overhead. In addition, a service composition problem involving multiple QoS metrics (as e.g. one additive and one concave metric) can already be considered as NP-Complete [45]. Therefore most schemes consider only one network-specific metric, or in case of two metrics use heuristics to reduce the complexity of the routing problems.

To summarize, the main drawback of this kind of schemes is the overhead introduced by the required periodic updates of QoS and service related information. As a consequence the size of addressable scenarios is limited with regard to number of nodes. On the other hand, since all required QoS information is collected proactively, the resulting service path calculation can be addressed directly after a request. Thus the schemes in this category in general have a short request response time.

Decentralized approaches as [40][33][39][44] and [13] address the service composition problem in a more reactive manner. As a consequence is it possible to address also large scale scenarios, since no central entity having global knowledge during the task of service overlay setup or for the discovery of valid processing chains is required. The approach proposed in this paper can be classified as decentralized. In fact we combine a DHT-based search for service components with on-demand QoS measurements. The actual measurements are used to verify that the QoS constraints of the requested composed service are not violated.

.

3. Service Overlays an Enabler for Autonomic Service Control

To establish overlay creation, maintenance and routing we start at the question: “How can the network provider take an active role in the provisioning of

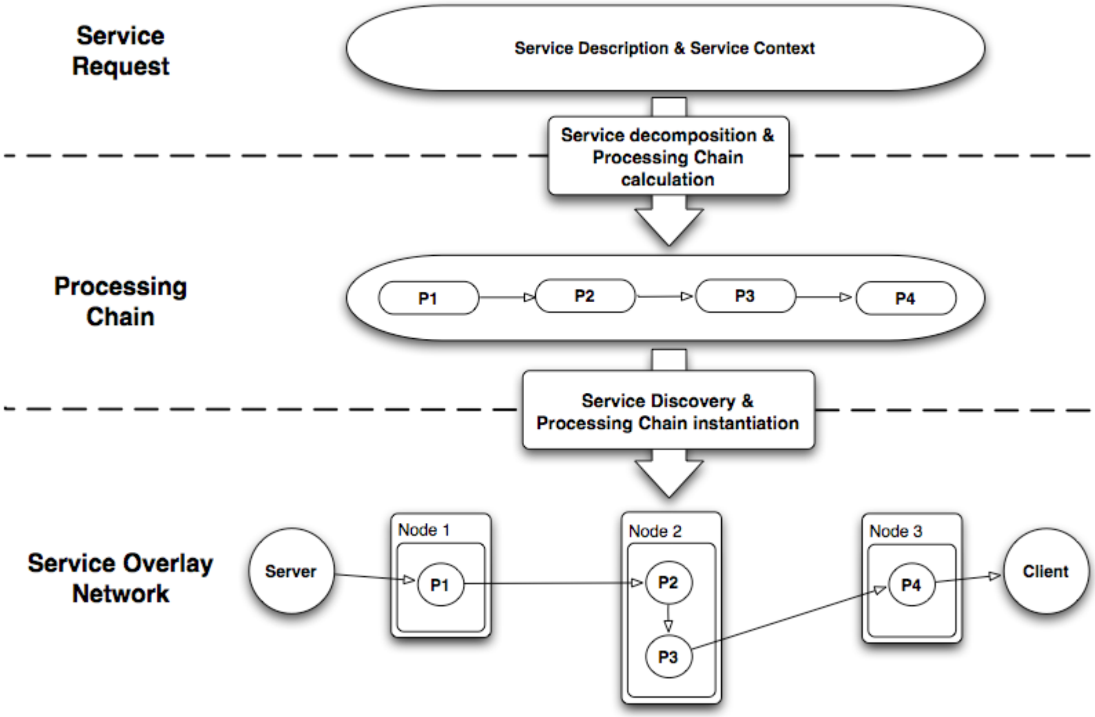

Service Request
Service Description & Service Context
Service decomposition & Processing Chain calculation
Processing Chain
P1
P2
P3
P4
Service Discovery & Processing Chain instantiation
Service Overlay Network
Server
Node 1
P1
Node 2
P2
P3
Node 3
P4
Client

Figure 2 Main Steps for Service Overlay Creation

services in future network environments?" In fact by integrating the Network Provider into the process of service provisioning, QoS related problems can be addressed cooperatively by interaction between the Service Provider, the Network Provider and the Client. The reason for this is the fact that in such a case all entities involved in transport of data related to a service are also aware of the service itself. As a side effect, a Network Provider can be part of the service value chain e.g. providing value added transport to third party service providers as well as its clients. As an expected positive impact such an approach will allow

1. The Service Provider to concentrate on Service/Content and to abstract from transport or end user terminal related issues.
2. The Network Provider to offer value added transport services as: Media Adaptation to client terminal as well as access technology, Broad/Multicast services, Caching.
3. The Client to access services that are optimised for his/her end user terminal as well as access network technology.

In the paper we will have a strong focus on Service Overlays for Media/data transport and adaptation services. The main reason for this is the fact that Media Services have stringent QoS requirements i.e. demand QoS aware transport and processing.

4. Service Overlay Creation

In the remainder of this paper we assume that a Service Overlay consists out of an ordered sequence of processing modules interconnecting a service source and sink. The main focus will be on how principles from the area of Peer-to-Peer (P2P) networks can be used to realize an autonomous overlay creation by using service specific self-configuration of a distributed system of processing modules, clients and servers. Before we describe the proposed strategy, Figure 2 show the two main steps required for the creation of Service Overlays in such a scenario.

Given a service description the first requirement is a methodology to decompose a given service request into a set of distributable sub services. In general there are two main ways to address this:

- *Online decomposition:* i.e. decomposition of service at time of request
-
- *Offline decomposition:* i.e. decomposition during registration of a new service (i.e. in advance of the first request).

In this paper we propose to focus on *offline service decomposition* using a Service Level Agreement (SLA) principle. The SLA has to be established between a service provider and a third party provider (e.g. Network Provider) in advance of the first service request. The main reason for the SLA based approach is low complexity compared to the requirement of using service description languages to formulate respectively parse service request. A second advantage of the SLA approach is the fact that both parties can proactively optimize their server or network infrastructure in advance of the first service request based on the expected amount of service users.

After receiving a request for a decomposed service it is required to locate nodes possibly distributed inside and/or at the edge of the network hosting processing modules required for the instantiation of the requested service. To accomplish this *Service Discovery* task we maintain the information about available processing modules inside or at the edge of the network, using a Distributed Hash Table (DHT) (e.g. [17][18]19]). Based on the result of this service location step it is now required to interconnect the service source and sink through a sequence of processing modules (PM) using an Overlay Network principle in a way that the QoS constraints of the service are not violated and the costs of service provisioning are minimized. In the remainder of the paper a PM is formalized as a triple of the Form (I, P, O) where:

- "**I**" refers to the possible input formats the PM can read (i.e. Layer II/III/IV specific)
- "**P**" refers to the processing function provided by the PM
- "**O**" refers to the output format the PM produces (i.e. Layer II/III/IV specific)

Since it is assumed that neither the sink (e.g. Media Clients (MCs)) nor the source of the dataflow (e.g. Media Servers (MSs)) do any processing they are formalized using just an **(I,O)** notation. The MC, requesting content from a MS, can be served directly if and only if the input "**I**" of the client is *compatible* to the output "**O**" of the Server. In the case of non-compatibility, a PM has to be inserted between the MS and the MC to realize the data delivery using a pipelining principle. To denote compatibility the symbol "~" is used.

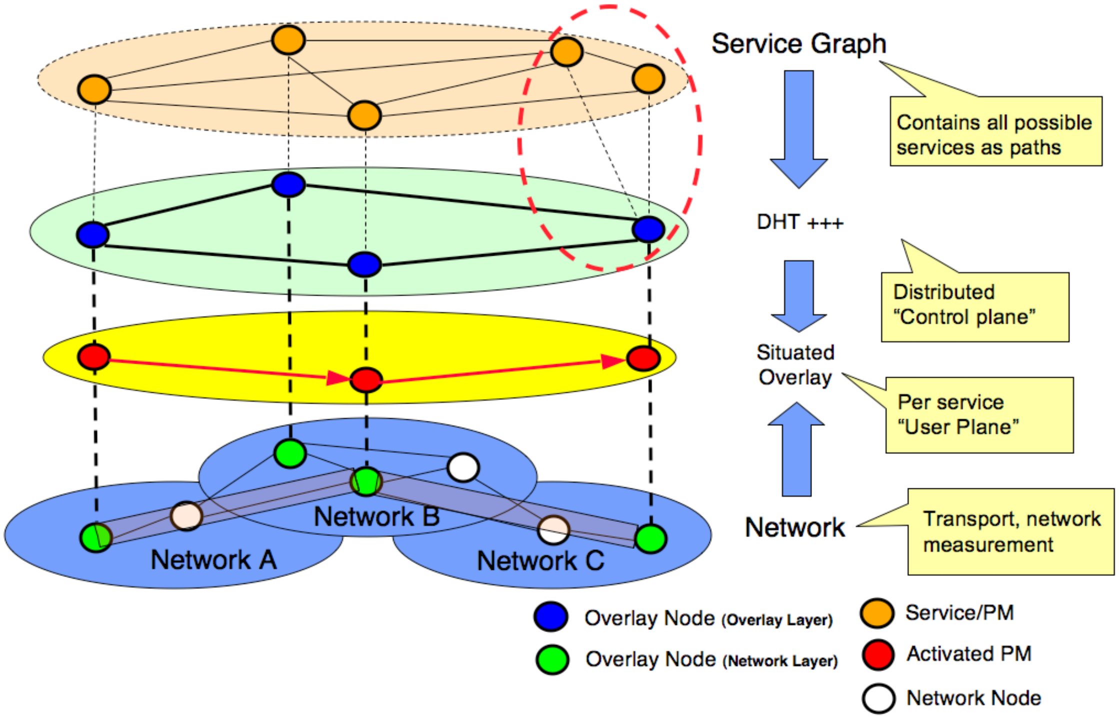

Service Graph
Contains all possible services as paths
DHT +++
Distributed "Control plane"
Situated Overlay
Per service "User Plane"
Network
Transport, network measurement
Network A
Network B
Network C
Overlay Node (Overlay Layer)
Overlay Node (Network Layer)
Service/PM
Activated PM
Network Node

Figure 3 Different System Levels

4. A DHT based control plane for Service Overlay Creation

In Figure 3 we show the different system levels involved in the proposed service overlay creation process. Starting from the top, the set of all possible services to be realized can be modeled in a graph structure called *service graph*.

Definition (Service Graph): Let

$$V = \{PM_1, PM_2, \ldots , PM_n\}$$

be a set of n processing modules. The Service Graph associated to V is defined as the graph SG(V, E) with

$$e = (PM_i, PM_j) \in E\text{:™ } PM_i < PM_j.$$

The property of a systems service graph we need in this paper is the fact that: *Every composed service that can be realized by a system corresponds to a path in its service graph*.

Using a problem specific indexing mechanism, the service graph structure is mapped into an address space of a Content Addressable Network (CAN) DHT [19]. This DHT will be extended towards a distributed control plane for the setup of service overlays.

The underlying idea of our approach is that in case every node that hosts processing functionality is also actively integrated into the search process, it is possible to build up the service overlay level while performing the search for its required processing functions. More concrete we are addressing Service Overlays creation based on a distributed CAN search principle combined with a hop-by-hop QoS constraint verification and propagation technique instead of extending classical routing algorithms as Dijkstra or Bellman-Ford (c.f. [20][21]). Using a DHT as the distributed control plane for a search & verify based approach has the following promising properties:

- *The resulting system can be realized in a fully distributed fashion and inherits the self-∗ properties of DHTs. Further it can be realized with comparable low management state per*

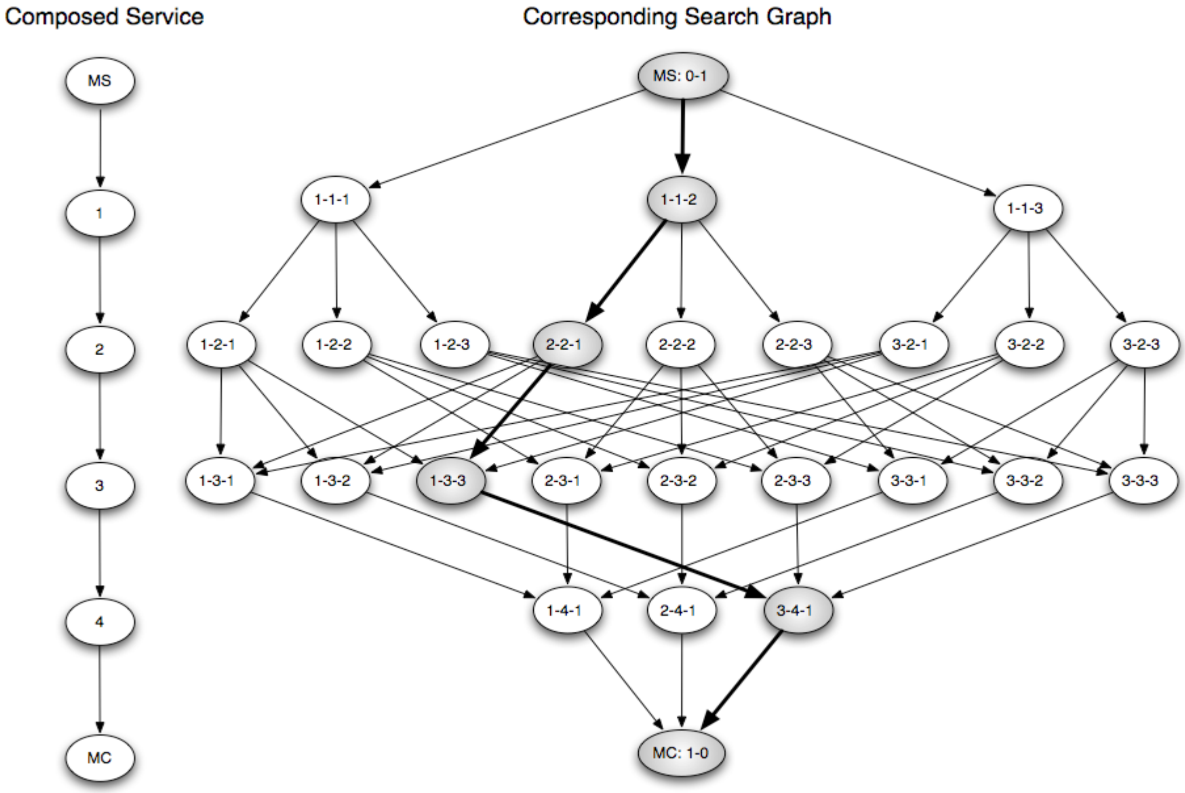


Figure 4 Search Graph

node e.g. O(log N), where N is the number of DHT nodes.

- *DHTs represent a well studied, resilient and fully decentralized domain for search based problems.*

5.1. Service Graph Embedding

As a prerequisite of the envisioned DHT based approach, it is required to specify how to embed a Service Graph structure into the corresponding DHT address space using a problem specific indexing scheme.

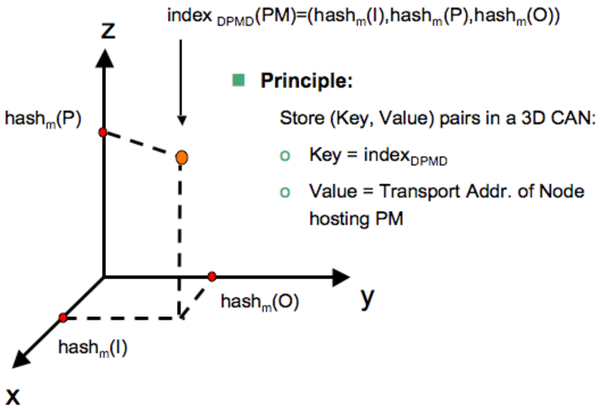

index $_{DPMD}$(PM)=(hash$_m$(I),hash$_m$(P),hash$_m$(O))
z
hash$_m$(P)
Principle:
Store (Key, Value) pairs in a 3D CAN:
Key = index$_{DPMD}$
Value = Transport Addr. of Node hosting PM
hash$_m$(O)
y
hash$_m$(I)
x

Figure 5 Indexing

To embed such a graph into a DHT address space we propose to focus on indexing functions that have the property that the above mentioned "~" relation is invariant with regards to the indexing process. We will illustrate this now by using a CAN DHT with address space $[0,t]\times[0,t]\times[0,t]\subset R^3$, for $t \in R$, t>0. After a new node n has joined successfully the CAN, the address where to store a pointer to the transport address where to find $PM_1 = (I_1 , P_1 , O_1)$ hosted by n, can be calculated as the coordinate

$$HASH_{CAN}(I_1,P_1,O_1) := (H(I_1), H(P_1), H(O_1))$$

for a hash function H having its values in [0,t] (c.f. Figure). The relation $\underset{\cdot}{<}$ is invariant with regard to $HASH_{CAN}$ since:

$$PM_i \underset{\cdot}{<} PM_j \text{™} HASH_{CAN}(PM_i) \underset{\cdot}{<} HASH_{CAN}(PM_j)$$

In case each new PM made available to the system is registering itself at a CAN using the described $HASH_{CAN}$ function, we can find a PM_2 offering the processing function P_2 while being compatible to PM_1 by forwarded the search to all nodes in the CAN address sub space ($H(O_1)$,$H(P_2)$, ∗) where "*" denotes any possible value.

5.2. Search & Verify

As stated before, every service that can be realized by a provider is corresponding to a path in its service graph. In case we want to realize a requested service it is required to find a corresponding path while taking the situation in the network into account. This task can be interpreted as a generalized Constraint Based Routing Problem (CBRP) including:

- a vector of QoS constraints related to the service
- a vector of constraints per Processing Step (e.g. delay introduced, costs etc.)
- a compatibility requirement between all the entities involved in service provisioning.

In its most general form such a CBRP can be formulated as:

Problem P1: Find an instance of the chain

$$(I_{MS},O_{MS}) \sim (I,P_1,*) \sim P_2 \sim \ldots \sim P_{i-1} \sim (*,P_i,O) \sim (I_{MC},O_{MC})$$

while the constraints $C_{SID} = (C_1,...,C_n)$ are fulfilled.

We will approach P1 using a Search & Verify principle to distribute and parallelize the search for a solution of P1 between the nodes forming the CAN DHT layer. The basic principle of the proposed Search & Verify approach is shown using a simplified example with only one PM one additive metric as delay and one concave QoS metric as bottleneck bandwidth in Figure 6.

After the media source received the request for the service with $PC_{SID}=(P_1)$ it initiates a search for a processing module able to accomplish P_1. For each match a verify procedure is started measuring the values $H_1=(h^1_1,h^1_2)$ between the source and the PM, to verify the QoS parameters associated with the service, which have been specified via C_{SID}.
If $h^1_1 <= C_1$ and $h^1_2 <= C_2$, the corresponding PM is starting a new measurement task between itself and the

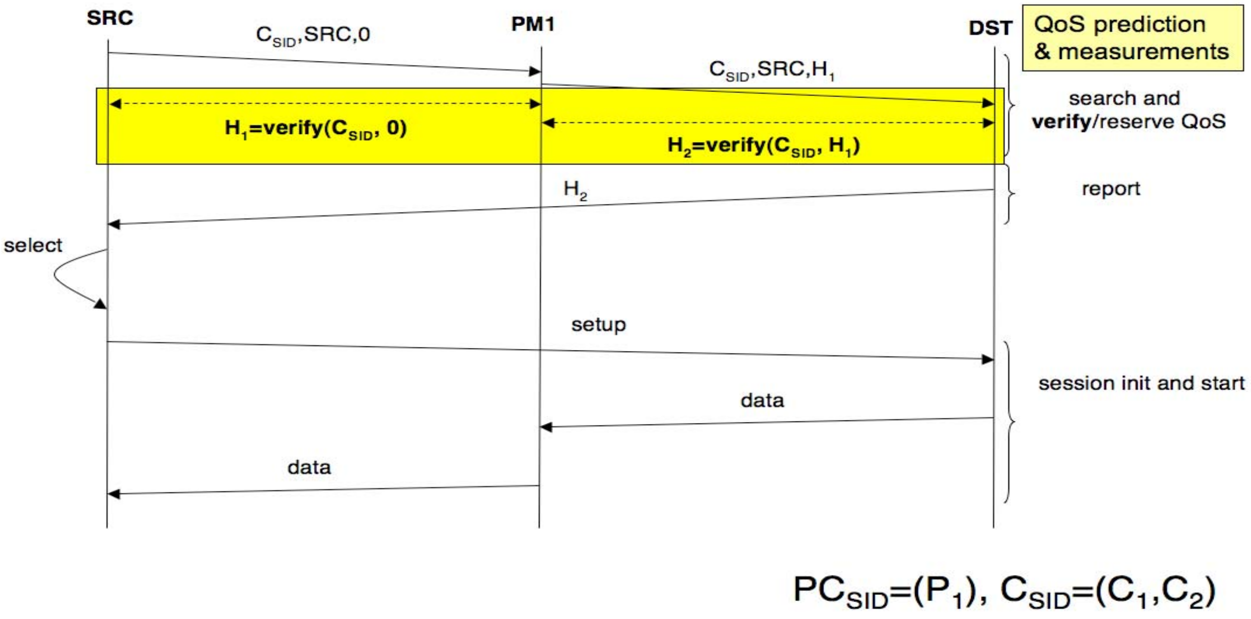


Figure 6 Search and Verify Approach

destination with result h^2_1, h^2_2. In case $h^1_1+h^2_1 <= C_1$ and $\min(h^1_2, h^2_2) <= C_2$ the destination is contacted and informed about the possible service chain found. In case not, all resources bound by the process are freed. In the simplified example, the client now reports all the possible service chains back to the source which selects the most adequate one based on QoS and cost values and initiates the data transfer.

In case of more complex services, the set of all possible processing chain candidates is in general defining a Directed Acyclic Graph (DAG) connecting the source and destination. We will call this DAG also the search graph associated with a service.

Figure 4 is showing an example search graph associated with the service P_{SID}=(1,2,3,4). For a more detailed study of the structure of search graphs and their relation to the complexity of the search and verify approach we refer to [13].

6. A Cooperative Service Provisioning Principle based on Service Overlays

The required interaction between Service Provider, Network Provider and Client for service provisioning based on the proposed system is divided into three phases, which are:

1. Service Registration
2. Service Request Processing
3. Service Delivery

In the following we will describe each phase in more detail.

6.1. Service Registration

During registration of a Service we establish a Service Level Agreement (SLA) regulating the QoS aware transport and processing requirements for a concrete service the Service Provider want to offer to a given Network Provider's customers. Since we focus on multimedia services we call this SLA from now on Multimedia Transport Service Agreement (MTSA). As stated before a Service Registration approach allows an offline decomposition of a service in advance of the first request.

The main steps to be performed during MTSA creation are based on our entity model and are illustrated in Figure 7 The minimum required set of information

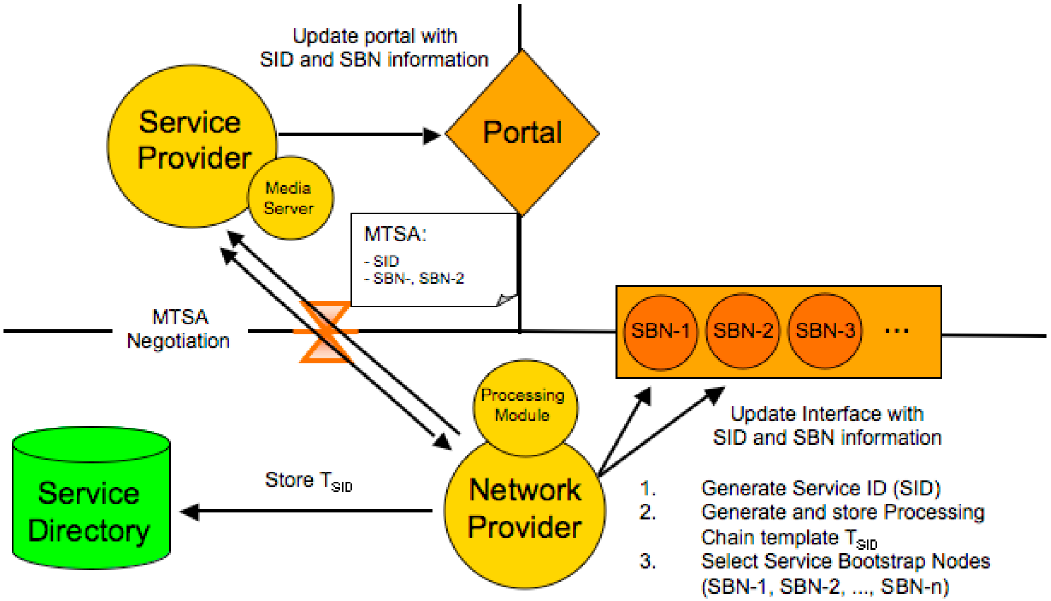


Figure 7 MTSA establishment

with regard to a new service to be negotiated during an MTSA agreement are:

- A unique Service ID (SID)
- A set of Service Sources (S_{SID}). E.g. a list of names or transport addresses of media servers hosting a special content
- A set of Service Bootstrap Nodes (SBNs) to initiate the Search & Verify process at the Network Provider.
- A set of required media/data processing steps (P_{SID}) to be performed by the Network Provider
- A set of constraints associated with the service (C_{SID}) as e.g. max. acceptable cost, max. acceptable loss or delay, bandwidth requirements etc.

We will use the term Processing Chain Template T_{SID} to denote the set of all information related to a service with id SID.

During a successful MTSA negotiation initiated by the Service Provider, the Network Provider is generating a SID and selects a set of Service Bootstrap Nodes (SBNs) to be used as entry points for accessing the new service. All this information is encapsulated into a T_{SID} which is stored at the responsible SBNs. After this step, the SID and the transport addresses of the responsible SBNs are sent to the Service Provider who can update e.g. a portal with information. From this point in time it is possible to access the service by connecting to a SBN responsible for the SID.

The decomposition of a service is done implicitly during the MTSA Negotiation process. To see this, we illustrate in Figure the main steps of an MTSA Negotiation. Essentially in Step 2 shown in the figure, the Network Provider is offering a set of service building blocks from witch the Service Provider is selecting a relevant subset at Step 3. After the selection the result is send to the Network Provider (Step 4). Therefore it is always guaranteed that the Network Provider can assign the list of processing steps P_{SID} to any negotiated Service SID.

6.2. Service Request Processing

Figure 9 shows the essential steps of a service request. From the portal of the Service Provider the client receives the required information to access a requested service i.e. the SID and the transport address of at least one SBN responsible for SID. With this information, the client can now send a service request to the SBN. As soon as the SBN has received the request, it performs a lookup operation for the corresponding T_{SID} extracting all information required to instantiate the service.

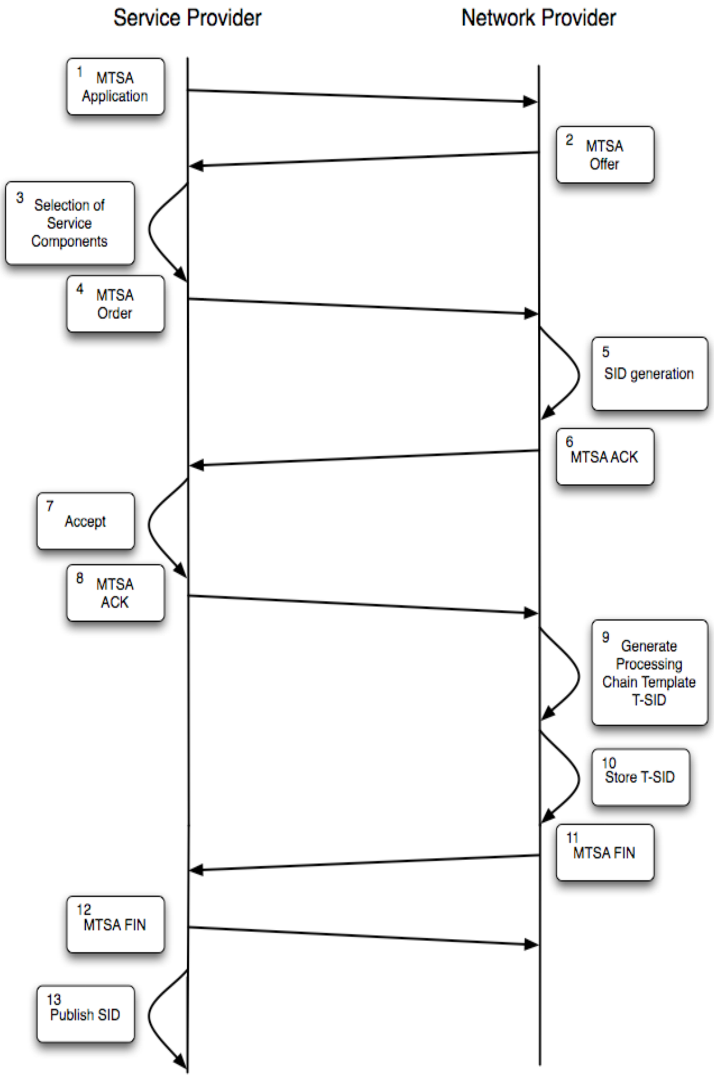


Figure 8 MTSA Negotiation

6.3. Service Bootstrap

At the point in time the Service Bootstrap is about to start the SBN has information about:

- **Service Specific**: Processing Chain $P_{SID}=(P_1,\ldots,P_i)$, Constraints $C_{SID}=(C_1,\ldots,C_n)$
- **Client Specific**: An acceptable input format I_{MC}
- **Server Specific**: At least one source with O_{MS}
- **Ingress PM Specific**: Required I and P_1
- **Egress PM Specific**: Required O and P_i

Based on this Information, the SBN can now initiate the creation of a Service Overlay network for SID. The SO will interconnect the Service Source, all required processing modules and the Client.

While all the former steps are in general based on the specification of an information model for service and network related data as well as the selection or design of suitable signaling protocols, this step is considered as the core problem to be addressed and was defined as P1.

7. Access Control in the Service Chain

One critical issue for a provider with regards to P2P based overlay technology is the fact that security and access control is often not an integral part of overlay networks [4]. A service platform without access control is incomplete. P2P research primarily perceived firewalls as an obstacle for the mutual connections between the participating overlay hosts [2][3] some overlays even disguise their traffic and tunnel through firewalls [16]. However, firewalls are successful security components that effectively guard services in the protected domain from unauthorized access. Packet filter firewalls are the predominant firewall architecture today. They base their decision on information from the headers of data packets, mainly at the network layer and the transport layer, but do not work well for hop by hop service chains. Traditional firewalls can only identify packets within the service chain that originate from the prior hop and are destined at the successor hop. They can neither identify the service source (the first service in the service chain) nor the service sink.

Firewalls loose their protective features if they cannot distinguish between legitimate and unauthorized overlay traffic. They will end up with a decision to either allow or to block all inbound overlay traffic.

The solution is to enhance firewalls for overlay networks to make them aware of service requestor, the

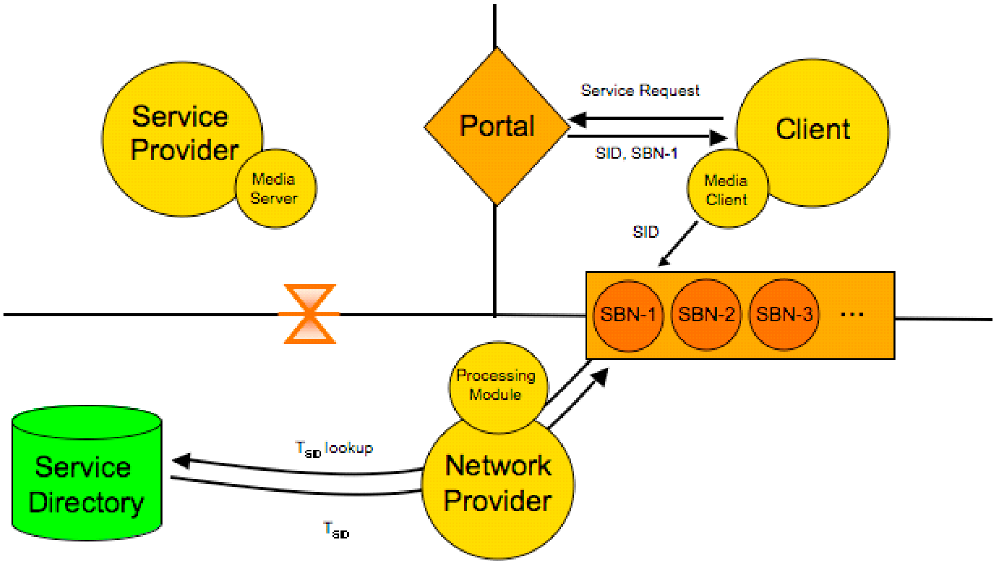


Figure 9 Service Request

processing modules and the destination of a service request. This new component must also be session aware to enforce fine grained access control decisions, for instance, a rule that a user is allowed to access a service only one time.

7.1 Overlay aware Access Control

Our approach is to extend the situated overlay by service chain aware access control functionality. There are two issues that require the integration with the overlay signalling.

1.) The authentication and authorization usually involves the cooperation with the service requester. Any service in a service chain might require authorization.

2.) Any access control function within the service chain must be aware of successor and predecessor and should also be able to distinguish and control traffic on a per session basis.

The integration of the overlay access control with the service chain instantiation solves these problems. The Situated Overlay discovers during the creation of a Service Overlay network for SID, that authorization is required at a number of services in the service chain. The Overlay supplies the access control functions with the next-hop and prior-hop address and describes the user session. Each MS, PM, and MC can either implement its own authorization policies, or rely on an authorization decision by another party (e.g. the network provider). Authentication deserves special attention in an overlay: Authentication is needed to verify subscriber identity.

7.2 Authentication

Authentication is responsible for asserting that the identity of the requester, as stated in the request, is correct. We give a brief overview how an overlay solution could integrate different NGN identity solutions for user authentication. NGN networks already have a sophisticated infrastructure for authentication in place (e.g. the Generic Authentication Architecture). The 3GPP defined with the TR 33.980 a coupling of the IMS with the Liberty Alliance - Federated Identity Management Framework. The coupling is done with the help of the Web Services Framework (ID-WSF) and the Federation Framework (ID-FF) for authentication. The 3GPP describes two options. The Network Application Function (NAF) is part of the IMS and can act as an Identity Provider (IdP) at the same time. The second option is to combine the NAF with the Authentication Service (AS) of the Web Services Framework. An overlay network can integrate these identity systems to obtain verified user identities.

8. P2P Principles for an Autonomic Service Control in the IMS

To be able to use P2P principles in the IMS context it is first required to address the question: What are

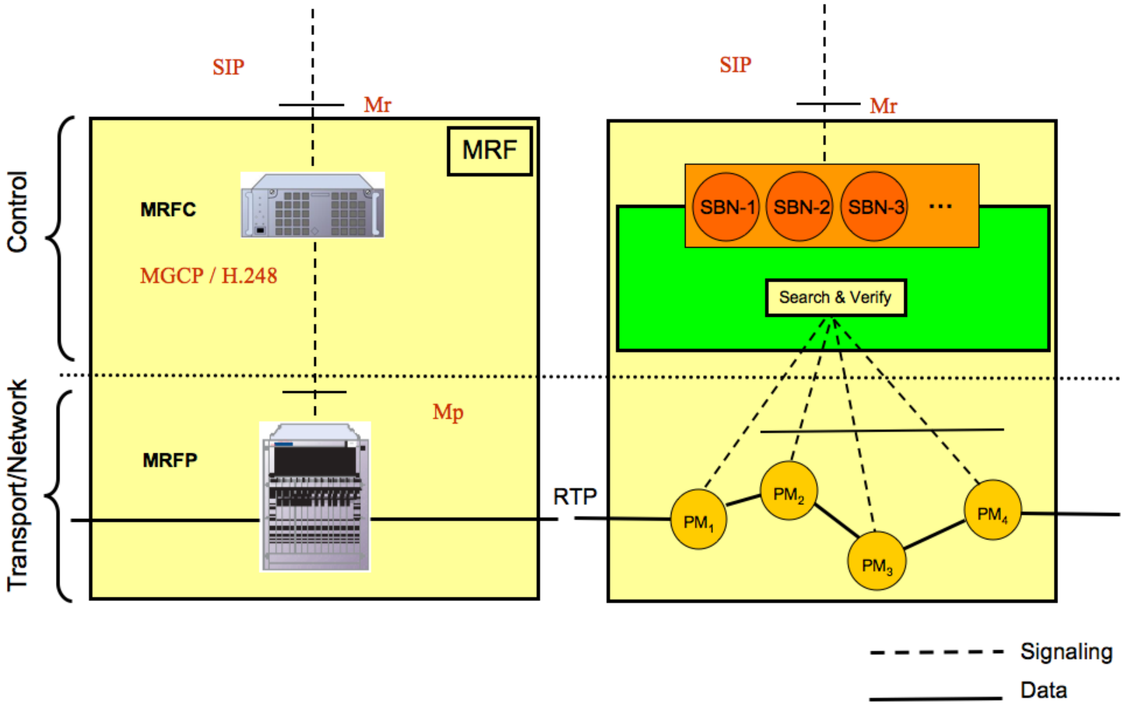


Figure 10 3GPP MRF

IMS functions that can be realised using P2P principles? As one candidate we identified the Media Resource Function (MRF) which is responsible for resource consuming tasks e.g. playing, transcoding and mixing of media streams. As an example from TISPAN, the Resource and Admission Control Subsystem (RCAS) realizes access and would profit from the proposed Situated Overlay approach by a better insight into service relationships for authorization decisions (refer to Section 7).

Another important issue is: What entities should run the corresponding P2P software? To address this, we categorise potential candidates into three main categories:

1. **Dedicated infrastructure nodes** are part of the network infrastructure and are particularly trustworthy and reliable.
2. **Home gateways and set top boxes** are fixed clients under partial control of the network provider that are already deployed in the order of several million devices in Europe.
3. **Mobile clients** have unsteady availability and would typically appear as service-originator or consumer in the situated overlay.

The IMS can be considered as a SIP based signaling overlay including AAA functions and support for multimedia calls and services. To enable the flexible introduction of new IMS based services a SIP Application server is included. To interface with an existing IMS it we propose the realization of an IMS Media Resource Function based on the described approach.

8.1. Realization of a distributed IMS Media Resource Function

The MRF represents the media related core of the IMS. It is composed out of the Media Resource Function Controller (MRFC) and the Media Resource Function Processor (MRFP). Since MRFC and MRFP are centralized entities, the same scalability and reliability problems as in case of a classical client/server approach for media delivery can be anticipated.

In this context the described approach can be used to realize a distributed MRF utilizing P2P principles. In addition it is possible to extend standard MRF functionalities as conferencing (mixing), recording and playback of static content (e.g. announcements) with additional services as

- Transcoding and/or downscaling of audio/video data,
- Traffic shaping to avoid congestion,
- Network based error correction techniques

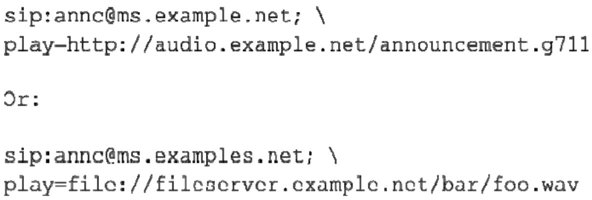
by registering the corresponding processing functionality in form of PMs.

In Figure 10 we mapped the relevant control as well as network/transport related functionalities to a 3GPP MRF.

As illustrated in the Figure 10 the Mr interface can act as the interface between an IMS and the SBN component. As a prerequisite for this it is required that an instance of the Mr interface is integrated into SBN nodes. In the reminder of the section, we use a SIP based approach for MRF control described in [22] to provide an example how such an Mr interface can look like.

The MRF control protocol described in [22] utilizes SIP INVITE or SIP INFO messages as carrier for service request from a SIP Application Server (SIP AS) or the S-CSCF to the MRF. The actual service request and the required parameters are encoded into the SIP URI [22] contains specifications how to control *announcements*, *interactive voice response* (Prompt and Collect) and *conferences*.

After receiving and processing a service request, the MRF is sending back a return code to signal successful completion or failure of the service request. To illustrate this approach Figure 12 shows two possible SIP request URIs for an *announcement.*



```
sip:annc@ms.example.net; \
play-http://audio.example.net/announcement.g711

Or:

sip:annc@ms.examples.net; \
play=file://fileserver.example.net/bar/foo.wav
```

Figure 12 RFC 4240 Service indicator for Announcements

Announcements are media files played to a user. Following the description in the RFC “Announcements can be static media files, media files generated in real-time, media streams generated in real-time, multimedia objects, or combinations of the above.” In addition the

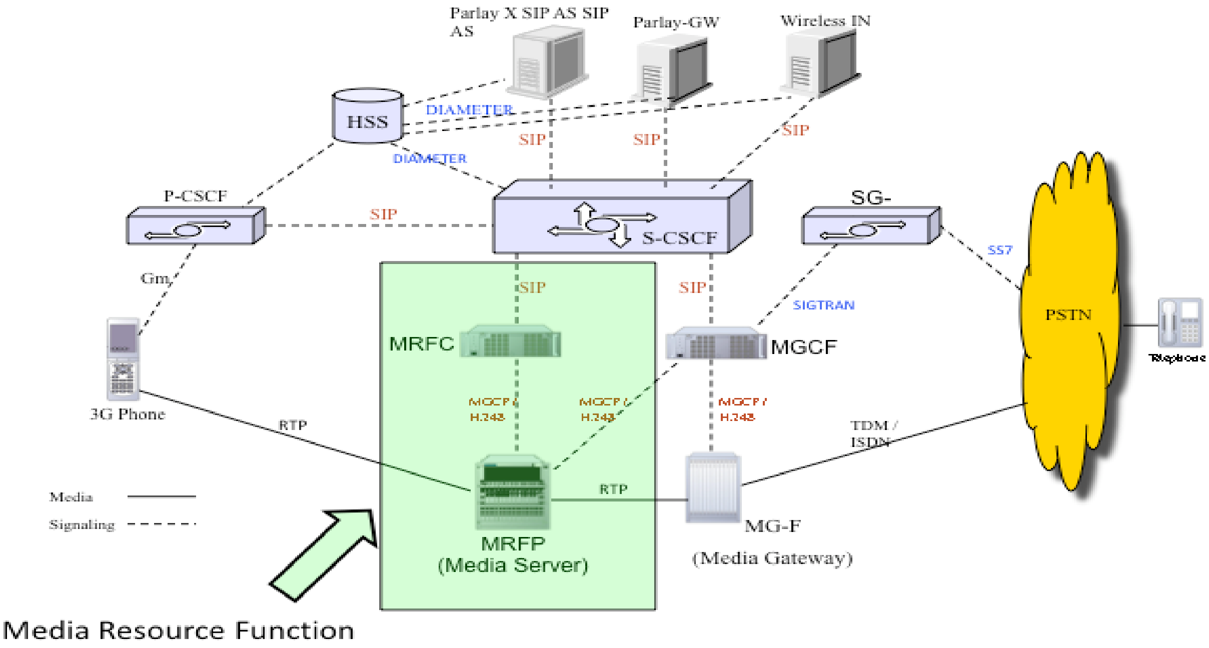


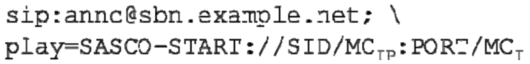
Figure 11 Media Resource Function in the IMS

authors state that “the mechanism described in this document has absolutely no impact to SIP devices other than media servers”.

To adopt the announce command to start and stop Topic O services we need to transport:

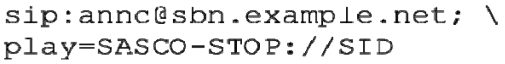
- The ID of the Service requested (SID)
- Client IP and Port
- The Capabilities of the requesting Client (MC$_I$)

As a consequence a start message send from SIP-AS or S-CSCF can be of the form:



sip:annc@sbn.example.net; \
play=SASCO-START://SID/MC$_{TP}$:PORT/MC$_T$

The corresponding stop message can be defined as:



sip:annc@sbn.example.net; \
play=SASCO-STOP://SID

Based on this categorisation we propose to use a hybrid P2P IMS approach using a combination of infrastructure nodes and fixed clients for a P2P based IMS since these nodes can be assumed to have the necessary stability and connectivity to be integrated into a service provisioning process.

To illustrate this Figure 11 shows how a distributed IMS MRF can be realised using the situated overlay and the IMS Mr interface to link the situated overlay to the rest of the IMS.

9. Conclusion

We are at the brink of the realization of Next Generation Networks with IMS at the core. To be able to manage the expected increasing configuration complexity caused by the plethora of future services we are convinced that self-management capabilities are crucial for the success of IMS/TISPAN based technology. To reach this we propose to exploit the autonomic functionalities of peer-to-peer based overlay technology to form an Autonomic Service Control for Next Generation Networks. We identified suitable entities that could form the P2P overlay network and IMS functions that benefit from a realization as overlay service. We introduced a decentralized service composition mechanism that obeys quality of service parameters. We argued with an access control scenario how service chain knowledge facilitates the required signaling for authorization of service usage. Future work is to realize the Autonomic Service Control and evaluate its role for IMS/TISPAN architectures.

10. Acknowledgements

This research is performed in the context of the Situated Autonomic Service Control (SASCO) project, which is gratefully funded by the France Télécom R&D.

11. References

[1] Andreas Klenk, Michael Kleis, Benoit Radier, Sanaa Elmoumouhi, Georg Carle, and Mikael Salaun, "Towards Autonomic Service Control in Next Generation Networks", The Fourth International Conference on Autonomic and Autonomous Systems, ICAS 2008, March 16-21, 2008 - Gosier, Guadeloupe

[2] Baset, S.A. & Schulzrinne, H., 'An Analysis of the Skype Peer-to-Peer Internel Telephony Protocol', Arxiv preprint cs.NI/0412017 (2004).

[3] Ennis, D.; Anchan, D. & Pegah, M.,The front line battle against P2P, in 'SIGUCCS '04: Proceedings of the 32nd annual ACM SIGUCCS conference on User services', ACM Press, New York, NY, USA, pp. 101—106 (2004).

[4] Wallach, D.S.,A Survey of Peer-to-Peer Security Issues., in 'ISSS', pp. 42-57 (2002).

[5] D. Xu and K. Nahrstedt, “Finding service paths in a media service proxy network,” Proceedings of the ACM/SPIE Conference on Multimedia Computing and Networking, (2002).

[6] K. N. Jingwen Jin, “Source-based QoS service routing in distributed service networks”, Proceedings of IEEE International Conference on Communications 2004 (ICC2004), 2004.

[7] X. Gu, K. Nahrstedt, and B. Yu.’SpiderNet: An Integrated Peer-to-Peer Service Composition Framework’,Proceedings of the thirteenth IEEE International Symposium on High-Performance Distributed Computing (HDPC-13) 2004

[8] P. Maymounkov and D. Mazieres, “Kademlia: A peer-to-peer information system based on the xor metric,” in Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS ’02), March 2002

[9] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, “A scalable content-addressable network,” in Proceedings of the 2001 ACM SIGCOMM Conference, August 2001.

[10] Ganek, A. G. Corbi, T. A.: The dawning of the autonomic computing era. IBM Systems Journal. 42(1), 5—18 (2003)

[11] J. H. Saltzer, D. P. Reed, Anind Dey, Understanding and D. D.Clark. End-to-end arguments in system design. ACM Transactions on Computer Systems, pages 277-288, 1984.using Context, Personal and Ubiquitous Computing, 2001

[12] O. Babaoglu, M. Jelasity, A. Montresor, C. Fetzer, S. Leonardi, A. van Moorsel, and M. van Steen, Eds., Self-Star Properties in Complex Information Systems, ser. Lecture Notes in Computer Science, Hot Topics. Springer-Verlag, 2005, vol. 3460.

[13] Michael Kleis, Kai Büttner, Sanaa Elmoumouhi, Georg Carle, Mikael Salaun, ”CSP, Cooperative Service Provisioning using Peer-to-Peer Principles” Proceedings of 2nd IEEE/IFIP International Workshop on Self-Organizing Systems (IWSoS), 2007

[14] Andreas Klenk, Frank Petri, Benoit Radier, Mikael Salaun, Georg Carle “Automated Trust Negotiation in Autonomic Environments” Proceedings of 2nd IEEE/IFIP International Workshop on Self-Organizing Systems (IWSoS), 2007

[15] “Telecommunications and Internet converged Services and Protocols for Advanced Networking” TISPAN; NGN Functional Architecture Release, ETSI ES 282 001 V1.1.1, 2005

[16] Androutsellis-Theotokis, S. & Spinellis, D., 'A survey of peer-to-peer content distribution technologies', ACM Comput. Surv. 36(4), 335—371 (2004).

[17] P. Maymounkov and D. Mazieres, “Kademlia: A peer-to-peer information system based on the xor metric,” in Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS ’02), March 2002

[18] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan, “Chord: A scalable Peer-To-Peer lookup service for internet applications,” in Proceedings of the 2001 ACM SIGCOMM Conference, August 2001, pp. 149–160.

[19] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, “A scalable content-addressable network,” in Proceedings of the 2001 ACM SIGCOMM Conference, August 2001.

[20] J. Jin, K. Nahrstedt. Source-Based QoS Service Routing in Distributed Service Networks. In Proc. IEEE International Conference on Communications 2004 (ICC2004), Paris, France, Jun. 2004.

[21] S Ralph Keller. Self-Configuring Services for Extensible Networks -- A Routing-Integrated Approach. PhD thesis, Computer Engineering and Networks Laboratory (TIK) ETH Zurich, 2004.

[22] E. Burger, J. Van Dyke, A. Spitzer, “RFC 4240, Basic Network Media Services with SIP”, December 2005.

[23] E. Shi, I. Stoica, D. Andersen, and A. Perrig. OverDoSe: A Generic DDoS Protection Service Using an Overlay Network.Technical Report CMU-CS-06-114, Carnegie Mellon University Computer Science Department, 2006.

[24] L. Subramanian, I. Stoica, H. Balakrishnan, and R. Katz. OverQoS: An Overlay based Architecture for Enhancing Internet QoS. In Proceedings of USENIX, 2004.

[25] David G. Andersen, Hari Balakrishnan, M. Frans Kaashoek, and Robert Morris. Resilient Overlay Networks. In Proceedings of Symposium on Operating Systems Principles, 2001.

[26] J. Liebeherr, M. Nahas, and Weisheng Si. Application-layer multicasting with Delaunay triangulation overlays . Selected Areas in Communications, IEEE Journal on, 20(8):1472–1488, Oct 2002.

[27] S. Gribble, M. Welsh, R. von Behren, E. Brewer, D. Culler, N. Borisov, Steven E. Czerwinski, Ramakrishna Gummadi, Jon R. Hill, Anthony D. Joseph, Randy H. Katz, Z. M. Mao, S. Ross, and Ben Y. Zhao. The Ninja architecture for robust Internet-scale systems and services. Computer Networks, 35(4):473–497, 2001.

[28] J. Liang, X. Gu, and K. Nahrstedt. Self-Configuring Information Management for Large-Scale Service Overlays . In Proceedings of IEEE INFOCOM, 2007.

[29] D. Xu and K. Nahrstedt. Finding service paths in a mediaservice proxy network. In Proceedings of the ACM/SPIE Conference on Multimedia Computing and Networking, 2002.

[30] J. Jin and K. Nahrstedt. Source-Based QoS Service Routing in Distributed Service Networks. In Proceedings of IEEE International Conference on Communications, 2004.

[31] J. Liang and K. Nahrstedt. Service composition for advanced multimedia applications. In Proceedings of SPIE/ACM Multimedia Computing and Networking Conference (MMCN), 2005.

[32] R. Keller. Self-Configuring Services for Extensible Networks – A Routing-Integrated Approach. PhD thesis, Computer Engineering and Networks Laboratory (TIK) ETH Zurich, 2004.

[33] X. Gu, K. Nahrstedt, and B. Yu. SpiderNet: An Integrated Peer-to-Peer Service Composition Framework. In Proceedings of 13th IEEE International Symposium on High Performance Distributed Computing (HPDC), 2004.

[34] B. Raman, S. Agarwal, Y. Chen, M. Caesar, W. Cui, P. Johansson, K. Lai, K. Lavian, S. Machira ju, Z. Mao, G. Porter, T. Roscoe, and Mukund. The SAHARA Model for Service Composition Across Multiple Providers. In Proceedings of International Conference on Pervasive Computing Zurich Switzerland, 2002.

[35] S. Ponnekanti and A. Fox. SWORD: A Developer Toolkit for Web Service Composition. In Proceedings of 11th World Wide Web Conference (WWW2002 Web Engineering Track), 2002.

[36] B. Mathieu, M. Song, A. Galis, L. Cheng, K. Jean, R. Ocampo, M. Brunner, M. Stiemerling, and M. Cassini. Self-Management of Context-Aware Overlay Networks for Ambient Networks. In Proceedings of 10th IFIP/IEEE International Symposium on Integrated Network Management, 2007

[37] X. Gu and K. Nahrstedt. A scalable QoS-aware service aggregation model for peer-to-peer computing grids. In Proceedings of the 11th EEE International Symposium on High Performance Distributed Computing (HPDC), 2002.

[38] X. Gu, K. Nahrstedt, R. Chang, and C. Ward. QoS-Assured Service Composition in Managed Service Overlay Networks. In Proceedings of IEEE 23rd International Conference on Distributed Computing Systems (ICDCS), 2003.

[39] X. Gu and K. Nahrstedt. Distributed multimedia service composition with statistical QoS assurances. Multimedia, IEEE Transactions on, 8(1):141–151, Feb. 2006.

[40] M. Wang, B. Li, and Z. Li. sFlow: Towards Resource-Efficient and Agile Service Federation in Service Overlay Networks. In Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS'04), 2004.

[41] S. Yamaoka, T. Sun, M. Tamai, K. Yasumoto, N. Shibata, and M. Ito. ResourceAware Service Composition for Video Multicast to Heterogeneous Mobile Users. In Proceedings of 1st International Workshop on Multimedia Service Composition, 2005.

[42] S. Choi, J. Turner, and T. Wolf. Configuring Sessions in Programmable Networks. In Proceedings of IEEE INFOCOM, 2001.

[43] B. Raman and R. Katz. Load Balancing and Stability Issues in Algorithms for Service Composition. In Proceedings of IEEE INFOCOM, 2003.

[44] X. Gu and K. Nahrstedt. Dynamic QoS-Aware Multimedia Service Configuration in Ubiquitous Computing Environments. In Proceedings of IEEE 22nd International Conference on Distributed Computing Systems(ICDCS), 2002.

[45] M. R. Garecy and D.S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman, San Francisco, 1979.

High-level Models of Software-management Interactions and Tasks for Gradual Transition Towards Autonomic Computing

Edin Arnautovic, Hermann Kaindl, Jürgen Falb, Roman Popp
Institute of Computer Technology
Vienna University of Technology
Vienna, Austria
{arnautovic, kaindl, falb, popp}@ict.tuwien.ac.at

***Abstract*—For making software systems autonomic, it is important to understand and model software-management tasks. Each such task contains typically many interactions between the administrator and the managed software system.**

We propose to model software-management interactions and tasks in the form of discourses between the administrator and the software system. Such discourse models are based on insights from theories of human communication. This should make them "natural" for humans to define and understand. While it may be obvious that such discourse models cover software-management *interactions*, we found that they may also represent major parts of the related *tasks*. So, these well-defined models of interactions and tasks as well as their operationalization allow their execution and automation. Based on this modeling approach, we propose a specific architecture for autonomic systems. This architecture facilitates gradual transition from human-managed towards autonomic systems.

***Index Terms*—Self-managing systems; autonomic computing; interaction modeling**

I. Introduction

Today's software systems are usually distributed and very complex. They have a large amount of parameters and possible configurations and it is crucial to satisfy their quality requirements such as performance, availability and security. Management of these systems includes tasks required to control, measure, optimize, troubleshoot and configure software in a computing system.

In order to automate software systems' management tasks, it is important to understand and represent them in some more or less formal way. Any such task contains typically many interactions between the administrator and the managed software system. We found that modeling these interactions facilitates understanding and specifying tasks as well. In order to make interactions easy to understand and to specify by humans, their specification should be on a high level.

Thus, we propose to model the software systems' management tasks in the form of *discourses* between the administrator and the system. Such discourse models are based on insights from human communication theories and provide specifications for tasks and their interactions; for the basic approach see [1]. We elaborate on it here and extend significantly both our task and interaction specifications (task and discourse metamodel) as well as our approach to communication content representation (management domain content metamodel).

Although autonomic computing is a challenging vision, truly self-managed systems are hard to achieve and not (yet) in wide-spread use. The processes and means for the transition towards this vision are still not sufficiently investigated. In order to address these issues, this paper presents an approach to gradual transition towards autonomic systems (an earlier sketch of this proposal can be found in [2]).

The core idea of our approach is that the same interaction specification is used both for management by human administrators as well as for autonomic management. More precisely, the same discourse model is used for the automated generation of user interfaces for human management as well as for the specification of the interactions between the autonomic manager and the managed system in the case of autonomic management. Whenever a management task is sufficiently understood and a related implementation available in the autonomic manager, managing this task can be handed over to the autonomic manager without changing its interaction specification in the discourse model. As a consequence, a smooth and gradual transition towards self-managed software systems will be facilitated, where the portion managed by human administrators becomes smaller and smaller.

In contrast to the major body of research on autonomic systems, this approach does not focus on designing and developing autonomic managers per se. However, the transition towards autonomic systems and supporting means seem to be equally important for their acceptance. Our approach contributes to the latter and is, therefore, complementary to work on improving autonomic managers.

The remainder of this paper is organized in the following manner. First, we provide some background on the human communication theories that we build upon. Then we present our running example, that we use to explain our approach to representing interactions and tasks in the form of discourse models. As an important part for the operationalization of such discourse models, we define their procedural semantics. Then we specify both our high-level autonomic architecture and our transition process from human-managed towards autonomic systems. Two case studies indicate the feasibility of our approach. Finally, we discuss related work.

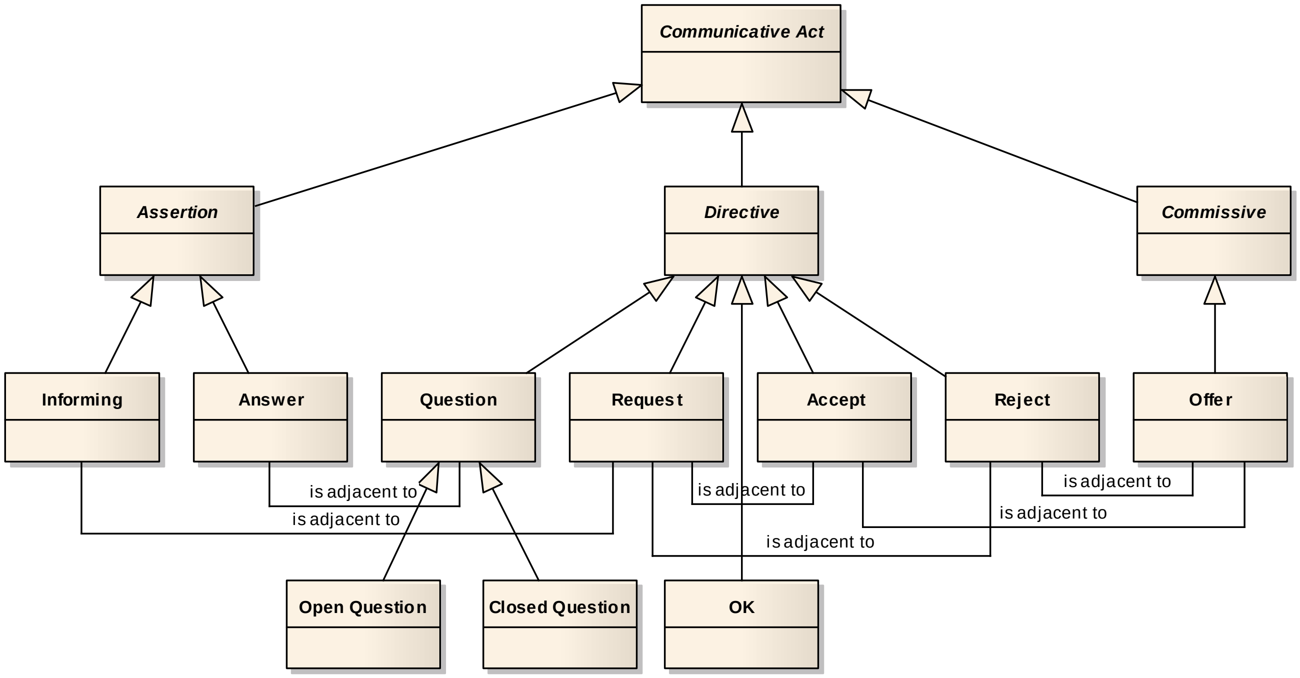


Fig. 1. Part of communicative acts hierarchy.

II. Human Communication Theories

Both tasks and their interactions can be specified in many ways. We strive for a uniform high-level approach to task and interaction representation based on the following human communication theories.

Communicative acts are derived from speech acts [3] and represent basic units of language communication. Thus, any communication can be seen as enacting of communicative acts, acts such as making statements, giving commands, asking questions and so on. *Communicative Acts* carry the intention of the interaction (e.g., asking a *Question*) and can be further classified into *Assertions*, *Directives* and *Commissives*. *Assertions* convey information without requiring receivers to act beside changing their beliefs (e.g., Informing and Answer). *Directives* (e.g., Question, Request, Accept) and *Commissives* (e.g., Offer) require an action by the receiver or sender for the advancement of the dialogue by further communicative acts. This classification is shown in Figure 1. The figure shows only a small selection from many communicative acts. Communicative acts have been successfully used in several applications: inter-agent communication in FIPA Agent Communication Language[1] (ACL), information systems [4] and high-level specifications of user interfaces [5].

Conversation Analysis. While communicative acts are useful concepts to account for intention in an isolated utterance, representing the relationship between utterances needs further theoretical devices. We have found inspiration in Conversation Analysis [6] for this purpose. Conversation analysis focuses on sequences of naturally-occurring talk “turns” to detect patterns that are specific to human oral communication, and such patterns can be regarded as familiar to the user. In our work we make use of patterns such as “adjacency pair” and “inserted sequence”.

Rhetorical Structure Theory (RST) [7] is a linguistic theory focusing on the function of text, widely applied to the automated generation of natural language. It describes internal relationships among text portions and associated constraints and effects. The relationships in a text are organized in a tree structure, where the rhetorical relations are associated with non-leaf nodes, and text portions with leaf nodes. In our work we make use of RST for linking communicative acts and further structures made up of RST relations. Thus, they represent the structure of possible interactions between an administrator and the software system. We use two types of RST relations: symmetric, multi-nuclear (e.g., *Joint, Otherwise*) and asymmetric, nucleus-satellite (e.g., *Result, Condition, Elaboration*).

III. Running Example

We use a simplified online store as our running example. The online store application enables the customer to look at and browse through different catalogues and products, to create user profiles, create and manage lists of preferred and desired products. It also allows ordering, shipping management and credit card processing. For the design of this application it is very important to separate data storage and management from data processing and presentation.

Such an application is usually implemented using a multi-tier architecture, with three as a typical number of tiers:

[1] Foundation for Intelligent Physical Agents, Communicative Act Library Specification, www.fipa.org

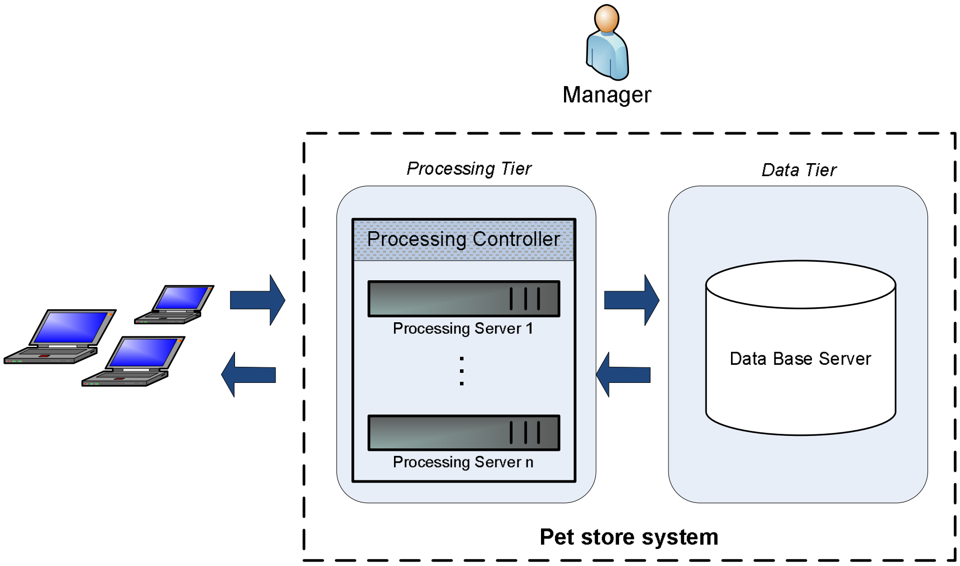


Fig. 2. Online store system architecture.

- presentation tier (e.g., implementing a Web interface),
- application logic tier (e.g., using Enterprise Java Beans), and
- data tier (e.g., using a relational database).

Yet, for our running example we integrate the presentational and logical functionality into one *processing tier*, and thus end up with a two-tier architecture for reasons of simplicity. The resulting online store architecture is shown in Figure 2. The data tier contains one database server. The processing tier contains a cluster with several processing servers, which are controlled by a *Processing Controller* server. A processing controller performs, for example, load balancing. The online store application is deployed in a hosting environment where each of the servers gets some amount of processing power assigned (e.g., using some virtualization technology). It is also possible to dynamically add additional servers and to integrate them into the processing tier. For each server in the architecture, the online shop owner has to pay a certain amount of money. The amount depends directly on the assigned processing power, which can be changed during runtime.

A major goal of the online shop owner is to achieve the best possible customer satisfaction and shopping experience. On the other hand, the owner wants to keep the operation costs as low as possible. One of the most important criteria for customer satisfaction in Web applications is the elapsed time from the page request until the requested page is fully displayed on the screen. This parameter is known as *response time*. Other parameters which influence user experience such as graphical design are not considered in our example. It is evident that the deployment architecture and the characteristics of the included components within this architecture directly influence the system response time. The manager of the shop application has the following possibilities to influence the runtime architecture and characteristics of the online shop:

- Increase or decrease assigned processing power of the database server.
- Increase or decrease assigned processing power of each of the processing servers in the processing tier.
- Add or remove a processing server in the processing tier.

The manager of the shop application can get the following information about the online store's state:

- average response time
- assigned processing power of the database server and of each processing server in the processing tier
- current utilization of the processing power of the database server and of each processing server in the processing tier
- average processing power utilization of the processing tier

The task of optimizing allocation of servers and their respective power to tiers in order to satisfy customers, and thus provide a high quality of service under peek loads and to keep the running costs low at the same time, is known under *provisioning*. Provisioning problems can be dealt with using complex mathematical models and architectures (e.g., according to [8]). Also some other information beyond the system itself can be required (e.g., current expense of the processing power per given unit). However, we do not go into more detail about such algorithms, since we are more interested in the communication which occurs within such management tasks.

In our running example, the manager of the shop application monitors the application's response time. If it happens to rise above a given limit, the manager tries to figure out more details about the cause, by acquiring information about the current runtime architecture, its structure and the properties of the system as a whole and its components. This includes, e.g., average processing power utilization in the processing tier, number of servers in this tier, assigned processing power and processing power utilization of each processing server as well as assigned processing power and processing power utilization of the database server. After having collected all this information, the manager decides to take some action: either to increase the assigned processing power of one of the servers or to add additional servers to the processing tier. When the manager realizes that the response time falls below a given threshold value, he can remove one of the servers from the processing tier to save operating cost.

For our approach it is important that the control and monitoring features listed above represent the content of the communication between the manager of the online shop system and the system itself. These features serve as the subject-matter that the manager and system are "talking about".

IV. High-level Interaction and Task Specification

We have developed a metamodel based on human communication theories which defines what the structure of the interaction and task models should look like in our approach. We explain it using an interaction specification for a management task based on the running example.

We model the communication between a managed software system and its (human or autonomic) manager in the form of discourses and relate them to the corresponding management tasks. Our conceptual metamodel is shown in Figure 3, specified as a UML class diagram.[2] While related concepts have been used for the modeling of human-computer interaction (e.g., in [9]), we use discourse models additionally for communication within software systems, more precisely between a managed software system and its autonomic manager. In addition to the communication specification, it is necessary to represent the communication content.

The metamodel illustrated in Figure 3 consists of two main parts. The upper rounded box represents discourses and management tasks. A management task represents a typical task of software system management such as system optimization, recovering from errors, etc. Figure 3 shows that a management task is specified by a discourse. A discourse consists of communicative acts, adjacency pairs, discourse relations, and their hierarchical structure and represents the modeling of interactions. This part is also used for general human-machine communication modeling [9][10][11].

The lower part of Figure 3 consists of classes involved in the description of the content to communicate for the purpose of the system's management and, therefore, represents elements of the management domain. There are two different kinds of information to be exchanged between a software system and its manager. These are the current state of the system captured by properties and management actions to be executed by the system.

A. Task and Discourse Metamodel

Our approach to task and communication models in the form of discourses can be sketched as follows. In essence, it has communicative acts as its "atoms", from which "molecular" structures can be composed in two dimensions. First, adjacency pairs model typical sequences of communicative acts within a dialogue that include turn-taking like for *question – answer* or *request – accept*. Second, *Discourse Relations* relate *Nodes* which can be adjacency pairs or other discourse relations, thus building up the hierarchical structure of the discourse. Discourse relations are further specialized into *RST Relations* (from Rhetorical Structure Theory) and *Procedural Constructs*.

The *Communicative Act* class represents a single interaction and carries the intention of the communication, e.g., asking a *Question* about the response time. Communicative acts can have many different types according to their communication intention as shown in Figure 1. The importance of explicitly specifying the intention is twofold. Raising the level of abstraction in general contributes to more "natural" specification and thus more efficient design. In our case, we raise the level of abstraction from simple messages (as e.g., in UML sequence diagrams) to communicative acts: questions, requests, offers, etc. In addition, we use the intention encoded in the type of the communicative act for automatic user interface generation [12]. For example, the intention of a *Question* communicative act is information gathering. Thus, input widgets like text areas or input boxes which allow the user to provide information are generated. The intention of the *Informing* communicative act is to provide new unknown facts to the receiver of the communicative act, where the receiver does not need to act upon receiving the communicative act. This can result in text or an image. Automatic generation of user interfaces would be much more difficult with interaction specifications on a lower level of abstraction (e.g., UML sequence diagrams).

Figure 4 shows an example discourse for optimizing the response time of the online store. The interactions depicted in the rounded boxes at the bottom of Figure 4 are cast in terms of communicative acts, e.g., the left-most *Question* for the system response time. The communicative acts in Figure 4 can be viewed as a usage scenario for optimization which advances from left to right. Since there are many sequences of interactions possible, we could think of this example more generally as a *use case*. While use cases carry additional information to sequences of actions, they barely represent more complicated structures.

More importantly, neither scenarios nor use cases represent something like intentions of the various interactions. Since our discourse models are built on communicative acts, they specify the type of communicative act for each interaction. E.g., Figure 4 shows *Questions* and their *Responses*, as well as *Requests* with *Accepts*. This extra piece of information carries the intent of such an interaction.

In addition, according to Conversation Analysis there are frequently occurring pairs of communicative acts — adjacency pairs. E.g., a *Question* must have a related *Response*, and a *Request* must have a related *Accept* or a *Reject*. Adjacency pairs can contain embedded dialogues like clarification dialogues which may become necessary before a communication party

[2] At the time of this writing, the specification of UML is available at http://www.omg.org.

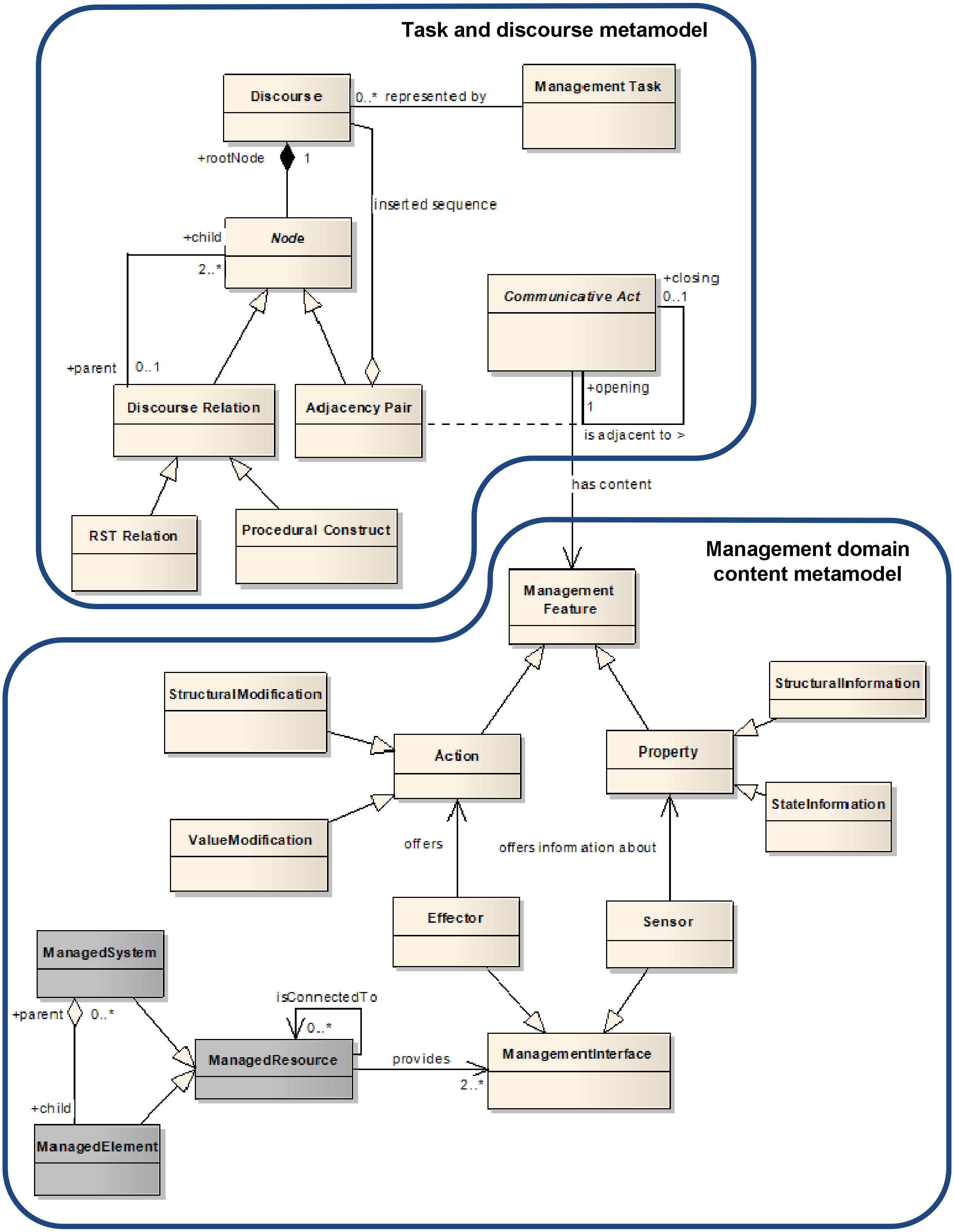


Fig. 3. Task and discourse metamodel connected with management domain content metamodel.

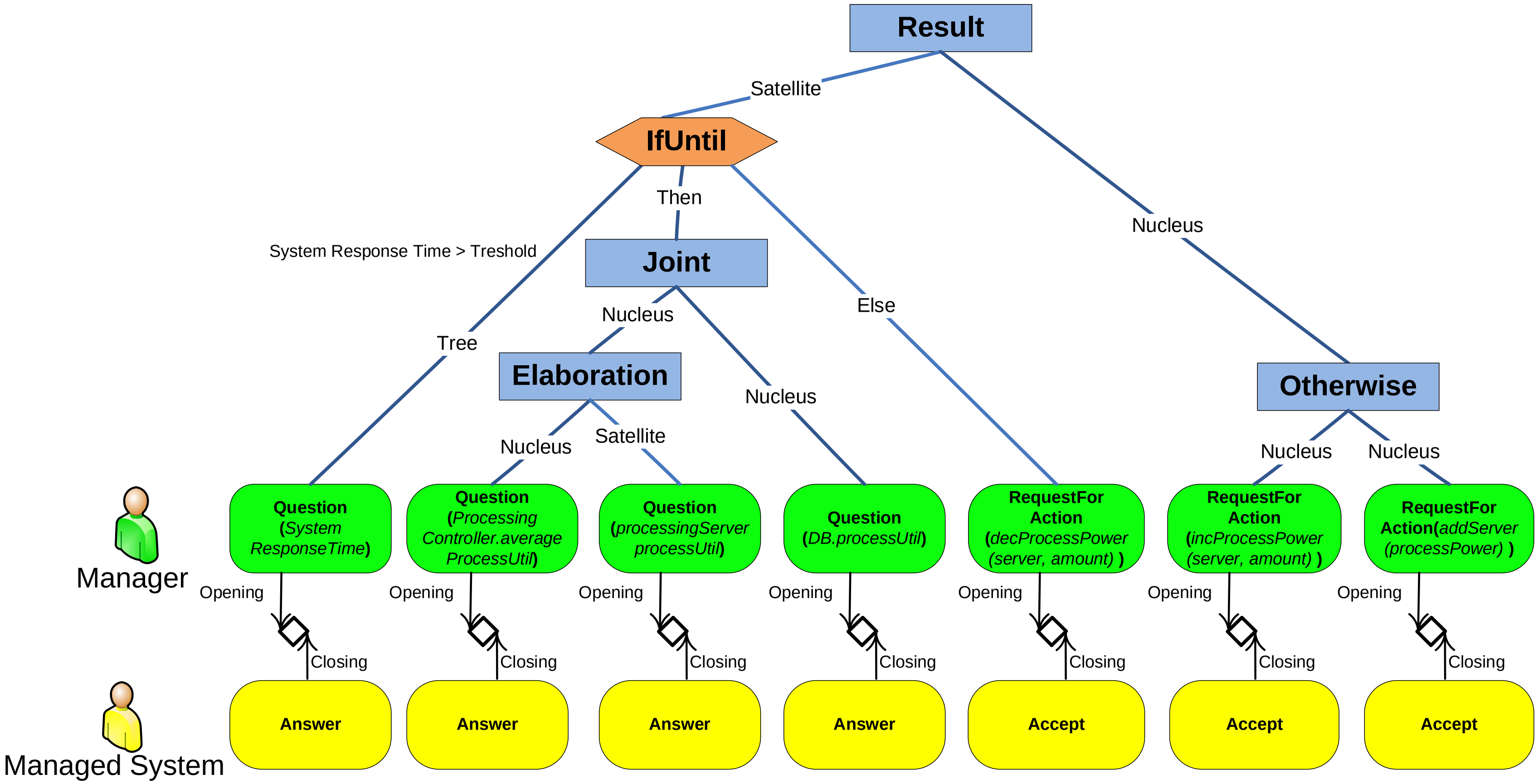

Result
Satellite
Nucleus
IfUntil
Then
Else
Tree
System Response Time > Treshold
Joint
Nucleus
Nucleus
Elaboration
Nucleus
Satellite
Otherwise
Nucleus
Nucleus
Manager
Managed System
Question (System ResponseTime)
Question (Processing Controller.average ProcessUtil)
Question (processingServer processUtil)
Question (DB.processUtil)
RequestFor Action (decProcessPower (server, amount))
RequestFor Action (incProcessPower (server, amount))
RequestFor Action(addServer (processPower))
Opening
Closing
Answer
Answer
Answer
Answer
Accept
Accept
Accept

Fig. 4. Discourse for optimizing the response time.

is able to answer a question, for example. Thus, adjacency pairs are modeled in our metamodel as association classes. In our example, adjacency pairs are graphically represented by diamonds. They connect the Opening communicative act with the Closing one.

As stated above, *Discourse Relations* relate adjacency pairs and further structures made up of Discourse Relations. They are specialized into *RST Relations* and *Procedural Constructs*.

All *RST relation*s used in our approach describe a subject-matter relationship between the branches they relate. They usually do not determine any particular execution order but eventually may suggest one. We use two types of RST relations: symmetric (multi-nuclear) and asymmetric (nucleus-satellite) relations. Multi-nuclear relations like *Joint* link equal discourse trees. In our example, the *Joint* relation links the Elaboration with questions about the utilization in the processing tier and a question about the utilization in the database. Note that the order of these actions is not specified by the *Joint* relation; that is why in this particular example also another scenario would fit in, where, e.g., first the database and then the processing tier is asked for its utilization. If it is possible, these questions could be asked even concurrently.

Nucleus-satellite relations link a discourse tree that represents the main intention and a discourse tree that supports the nucleus. For example, the *Elaboration* states that the satellite branch elaborates the dialogue executed in the nucleus branch. In Figure 4, asking the questions about the utilization of each server within the processing tier is the elaboration of the question about the average utilization in the processing tier (controller).

In addition to RST Relations, it turned out to be useful to be able to prescribe particular sequences and repetitions eventually based on the evaluation of some conditions. RST relations are not sufficient for this purpose and, therefore, we have introduced *Procedural Constructs* into our tree structure.

Procedural Constructs provide means to express a particular order between branches of the discourse tree, to specify repetition of a branch and to specify conditional execution of different branches. Thus, our procedural constructs add control structures to our discourse trees that are more complex than usual *if-then-else* or *repeat-until* constructs in typical procedural programming languages. When operationalizing the discourse tree, these procedural constructs also determine which information cannot be presented together on one screen of a graphical user interface. One such construct is *IfUntil*. E.g., in our example the *IfUntil* relation requires information about the server response time, and the execution continues only if the server response time exceeds some defined threshold.

Figure 4 represents a discourse for optimizing the response time for our running example and depicts such a tree structure of the discourse. It illustrates how all interactions within the discourse conceptually belong together as a whole. This structure is composed from Discourse Relations where RST relations are shown in boxes and the *IfUntil* procedural construct in a hexagon.

In our example, there is the relation called *Result* at the top. It represents that the actions requesting the increase of the processing power of a server *or* adding a new server to the system, are a consequence of the situation *resulting* from the preceding interactions. These preceding interactions are subordinated to the *IfUntil* procedural construct. After the Question about the system response time has been asked, the manager decides if the *Tree* branch has to be repeated, or either the *Then* or the *Else* branch will be executed. As stated above, a *Joint* relation (here within the Then branch) does not prescribe the order of execution and allows concurrency. The *Elaboration* relation relates the communication about the general properties and their details. A more formal specification of the relations is given below in Section V.

Such a tree of Discourse Relations could be viewed as the design rationale of the interactions. Alternatively, it can be viewed as a "plan" structure of the discourse for arriving at some goal. In this view, it is actually a non-linear plan (see the *Joint* relation in this example), while the usage scenarios are related linear plans. It is important to note that, where the discourse model represents a generic set of possible discourses, the concrete discourse flow will be controlled by the (human or autonomic) manager.

B. Management Domain Content Metamodel

Besides representing the communication flow explained above, a complete communication representation includes the representation of the communication content as well. The lower part of Figure 3 shows the part of the conceptual metamodel which describes the discourse content for managing software systems.

There are two different kinds of information to be exchanged between a software system and its (autonomic or human) manager:

- the information about current system *properties* and
- the *actions* to be executed by the system as requested by the manager.

Each communicative act is associated with its content: system properties and actions. *Properties* and *Actions* are generalized into the *Management Feature* concept in our metamodel. We distinguish between two types of properties: *StateInformation* and *StructuralInformation*. *StateInformation* represents the managed resource as seen from outside by its parameters (black box). *StructuralInformation* carries the information about the runtime architecture and structure of the system. Analogously, two types of Actions exist: *StructuralModification* and *ValueModification* for modifying the runtime architecture of the system or some of its properties.

Properties and Actions make up the *Sensor* and *Effector* interfaces, respectively, and are generalized into the *ManagementInterface* concept. Each of the *Managed Resources* is related to such management interfaces. Usually, a managed resource will contain one *Effector* and one *Sensor* interface, each containing several system actions and properties. However, a managed resource could have several of them, if interfaces group properties and actions according to some criteria (e.g.,

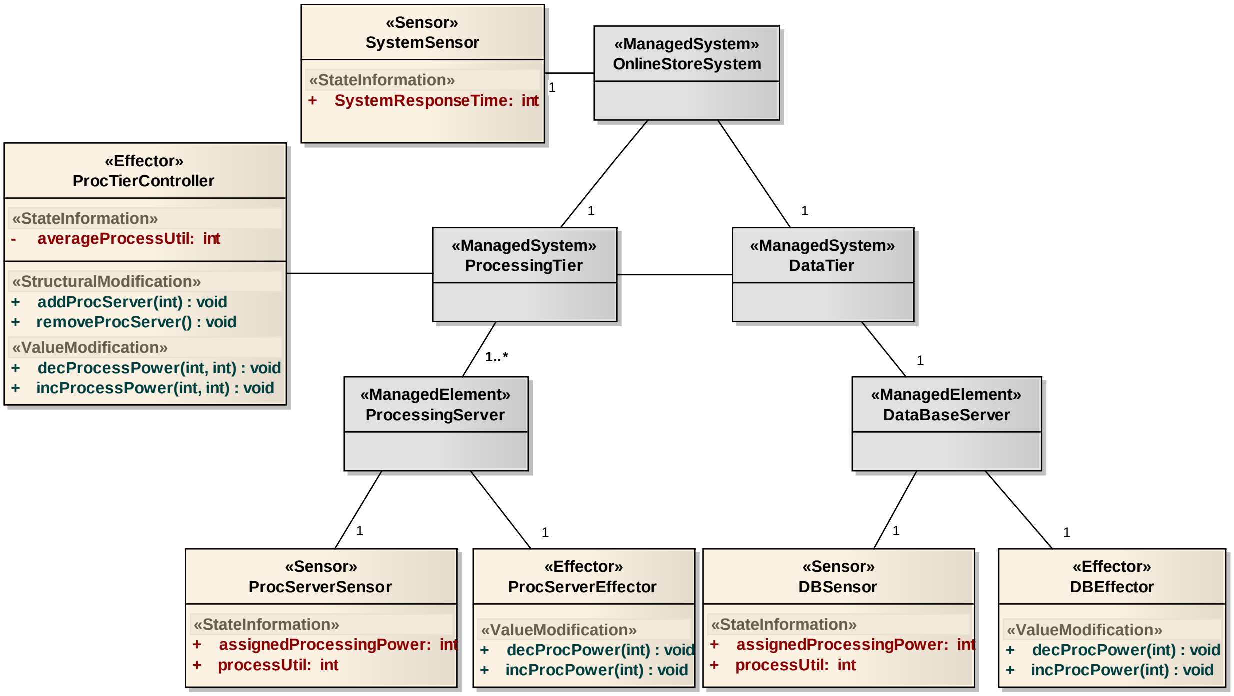


Fig. 5. Content representation for our running example.

one sensor containing only performance and another only security properties).

These management interfaces are provided by the *Managed Resource*, which is specialized into *Managed Element* and *Managed System*. Managed systems can basically contain other managed resources, thus allowing one to build a hierarchical structure of a managed system. These three concepts (dark gray classes in the metamodel in Figure 3) enable the modeling of the system structure, so that the manager and the managed system can communicate about it. The runtime instantiation of this model can be also used by the autonomic manager for reasoning. Structural properties and actions for changing the system structure are only possible for managed systems and not for managed elements.

Figure 5 shows an instantiation of this part of the metamodel for our running example and represents the model of the system architecture. The figure shows the model as a UML class diagram, where classes, methods and attributes are assigned with stereotypes corresponding to the metamodel. The gray classes represent the system structure of the *OnlineStoreSystem*, including two tiers and servers within these tiers. The pink classes represent the management interfaces. For example, *SystemSensor* is a management interface for getting the server response time and *DBEffector* for increasing and decreasing the processing power of the database server. Runtime architecture would be an instantiation of this model, where we would, for example, have several instances of the processing server.

V. Procedural Semantics

An important issue is to operationalize our communication and discourse specifications and to make them executable within our architecture. In order to achieve this operationalization, we transform the communication specifications in the form of discourses into state machines. In this sense, the transformation to state machines defines procedural semantics of our discourse models. For this transformation, the intentions of the communicative acts are not used, but they are not lost, since they are used for other purposes.

In many real-world applications, predetermined discourses are well suited for communication between human and computer, since the user can anticipate the behavior of the system and can expect the same user interface whenever starting the discourse with the system again. This behavior can be achieved by using state machines for the discourse management. Many approaches directly utilize some kind of state machine to model communication. These include both human-machine communication and user interfaces required for it (e.g., in [12][13]), as well as machine-machine communication (e.g., in [14]). Contrary to these approaches, we do not use state machines for explicit communication modeling, but solely for the specification of their procedural semantics and for their execution. Importantly, we do not let the user model them.

Since, especially in user interfaces, the possible interactions are manifold, a flat state machine can become quite complicated. A solution for reducing the complexity of state machines are statecharts, which introduce hierarchies into state machines. Since our discourse models are hierarchical as well,

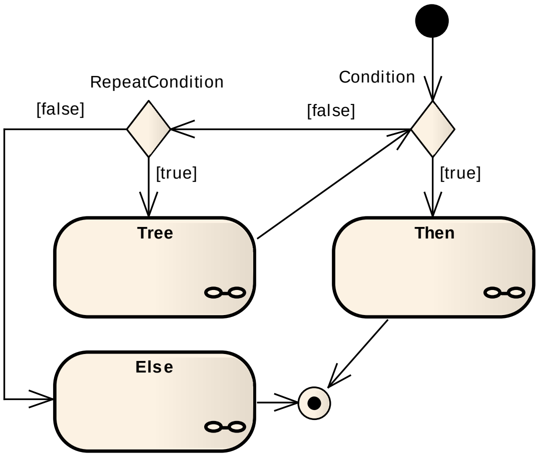


Fig. 6. Mapping of the *IfUntil* procedural construct.

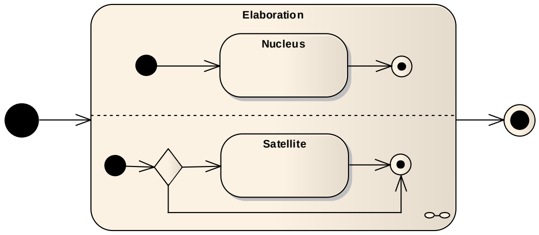


Fig. 7. Mapping of the *Elaboration* RST Relation.

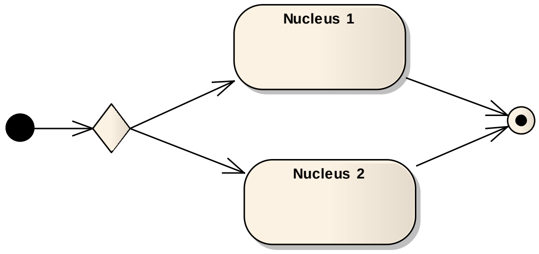


Fig. 8. Mapping of the *Otherwise* RST Relation.

we use statecharts to specify the procedural semantics of the discourse relations and thus the dynamics of the complete discourse.

Such statecharts have the following basic structure:

- Each state transition corresponds to exactly one communicative act and thus represents the advancement in the dialogue.
- State transitions are triggered by sending a communicative act by either the manager or by the managed system.
- Each state entry gets processed, resulting in system effects and in possible triggering of a new communicative act.
- Adjacency pairs are reflected in the statechart by a sequence of transitions. Thus, they constrain the set of potential communicative acts of outgoing transitions of the current and adjacent states.
- Discourse relations are mapped to state machine patterns forming a submachine state that can be included in higher-level discourse relation mappings.

In the following, we specify the procedural semantics of the discourse relations used in our running example.

The statechart of our *IfUntil* [11] relation is shown in Figure 6. *IfUntil* is a procedural construct that we found useful for defining a certain control structure in our task and discourse models. It is more complex than the usual procedural statement in a typical procedural programming language. In fact, it may be considered a combination of an if-statement and a conditional loop. If Condition is fulfilled in the statechart, the discourse continues in the Then branch.

Otherwise, there are two possibilities:

1) the Tree branch is performed if the Condition is not fulfilled. It can be performed again and again until Condition becomes fulfilled, or
2) the discourse can continue in the Else branch, if RepeatCondition is also not fulfilled.

This branching of the flow is modeled by the UML "choice" construct — graphically represented by a diamond. Checking, whether Condition and RepeatCondition are fulfilled is performed by the (autonomic or human) manager.

The *Elaboration* RST relation states that the satellite branch elaborates on the communication executed in the nucleus branch. The procedural semantics are defined in the statechart shown in Figure 7. Communication in the satellite branch is optional — the communication in the nucleus branch does not have to be elaborated. However, if the communication in the nucleus branch gets elaborated, it can even occur in parallel. This is decided usually by the manager for requesting additional information about some parameter. To complete the execution of the relation, both, nucleus and satellite have to be completed.

If the communication flow requires that in one particular moment either the one *or* the other branch of the discourse has to be performed, we use the *Otherwise* RST relation. Usually the manager decides which one of the nuclei (even choosing out of several ones) has to be performed. After one branch is started, it also has to be completed in order to complete the relation as a whole. The procedural semantics is shown in the statechart in Figure 8.

The *Result* RST relation represents that the communication in the nuclei is a consequence of the situation *resulting* from the communication taken place in the satellite. Its simple statechart is shown in Figure 9. It can be used for improving the rendering of the management user interface.[3]

The *Joint* RST relation is a multi-nuclei relation, which doesn't prescribe any order of the execution of the communication in its nuclei. The communication within nuclei can even be performed concurrently, and the relation is completed after all nuclei are completed. This possible concurrency is shown in its statechart in Figure 10 by two compartments separated by a dashed line.

[3] We have slightly changed its semantics with respect to [1].

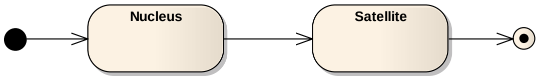


Fig. 9. Mapping of the *Result* RST Relation.

For operationalizing the complete discourse model, the statecharts of each relation have to be combined into one hierarchical statechart according to the discourse relation hierarchy in the discourse model. This is done by traversing the tree structure recursively and applying statechart mappings of the corresponding discourse relations. Typically, the statechart of one discourse relation is included as a submachine state in the statechart of the higher-level statechart. Therefore, the hierarchy of the overall statechart corresponds to the hierarchy of the discourse tree. The result of this traversing for our running example is presented in Figure 11. It completely defines the set of possible discourse flows.

Starting at the top of our example discourse in Figure 4, we have the *Result* RST relation. It basically defines two subsequent statecharts of the relations *IfUntil* and *Otherwise*. The *IfUntil* relation contains three branches: Tree, Then and Else. The Tree and Else branches don't contain any further discourse relations. Both of them contain one adjacency pair each, which are also mapped to the (simple) statecharts. The Tree branch contains the adjacency pair for acquiring system response time. The adjacency pair is mapped to the statechart containing two states: S1 and S2. The transitions within this statechart are induced by the communicative acts: "M: Question (System Response Time)" uttered by the Manager (**M**:) and "S: Answer" uttered by the System (**S**:). The Then branch contains the Joint relation, which again contains an Elaboration relation in one of its nuclei. The *Otherwise* relation doesn't contain any further discourse relations in its nuclei and the mapping defined above is applied, where each nucleus contains one adjacency pair. Prior to each state transition, either the manager or the managed system is supposed to fill in the content of the communicative act. For the decision points (e.g., in *IfUntil* or *Otherwise* relations) the manager is involved and it controls the further discourse flow. Such a complete discourse statechart is interpreted by the Task Execution Engine of our architecture.

VI. Autonomic Architecture and Transition Process

Figure 12 illustrates our proposed autonomic architecture designed to execute and automate modeled tasks as presented above. It is based on the generic autonomic architecture [15], separating the autonomic manager from the managed system.

The *Task Execution Engine* interprets the task and its associated discourse according to the procedural semantics of the discourse relations presented above and manages the flow of the communication. It utilizes also the intention encoded in the type of the communicative acts. For example, for a *Question* it would invoke some information gathering facility of the underlying managed system and for a *Request* it would call corresponding action. The *Human Administrator* or the *Autonomic Manager* controls the autonomic administration process by deciding which set of actions (set of communicative acts) has to be performed next.

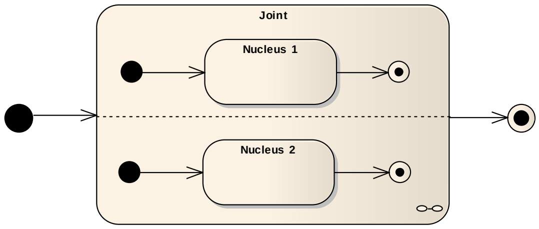


Fig. 10. Mapping of the *Joint* RST Relation.

The *UI Generator* generates and controls the user interface for the case of human management. It utilizes the procedural semantics of the subject-matter relationships of the RST Relations (e.g., Elaboration, Otherwise, etc.) as presented above. In addition, it takes into account the intentions of the communicative acts as mentioned in Section IV-A. The *Communication Adapter* encapsulates low-level interaction interfaces of the managed system.

This architecture enables the whole spectrum of management possibilities. Without an Autonomic Manager first, it provides for high-level communication with the human administrator only, through a generated user interface. In Figure 12(a), the human administrator is responsible for the management Tasks I and II. For example, management Task I could be the performance optimization as in the running example, and Task II could be a (self-)healing task.

After performing the management by humans for some time according to the defined discourse, it is expected that some insights into system behavior will have been gained. With these insights, the Autonomic Manager has to be designed and implemented. Since the discourse constrains possible interactions, this is easier to do than implementing the autonomic manager from scratch. Some of the relations include conditions for their execution as defined above, which have to be evaluated by the Autonomic Manager. It "decides" how the discourse proceeds by initiating the sending of corresponding communicative acts. Depending on the discourse complexity as well as the number and variety of relevant parameters and possible actions, the manager would be more or less complex. However, the design and deployment of autonomic managers is out of the scope of this paper.

VII. Case Study: System Simulation with SimSys

In the first cases study, we simulated an IT system with a tool and modeled one typical management task for it. We concentrated on human management using graphical user interfaces. The used tool has been developed in the course of a research project at IBM [16] and was kindly provided to us for our research. It has been used at IBM for the evaluation of policy-based software system management. This tool can simulate IT systems with different configurations containing

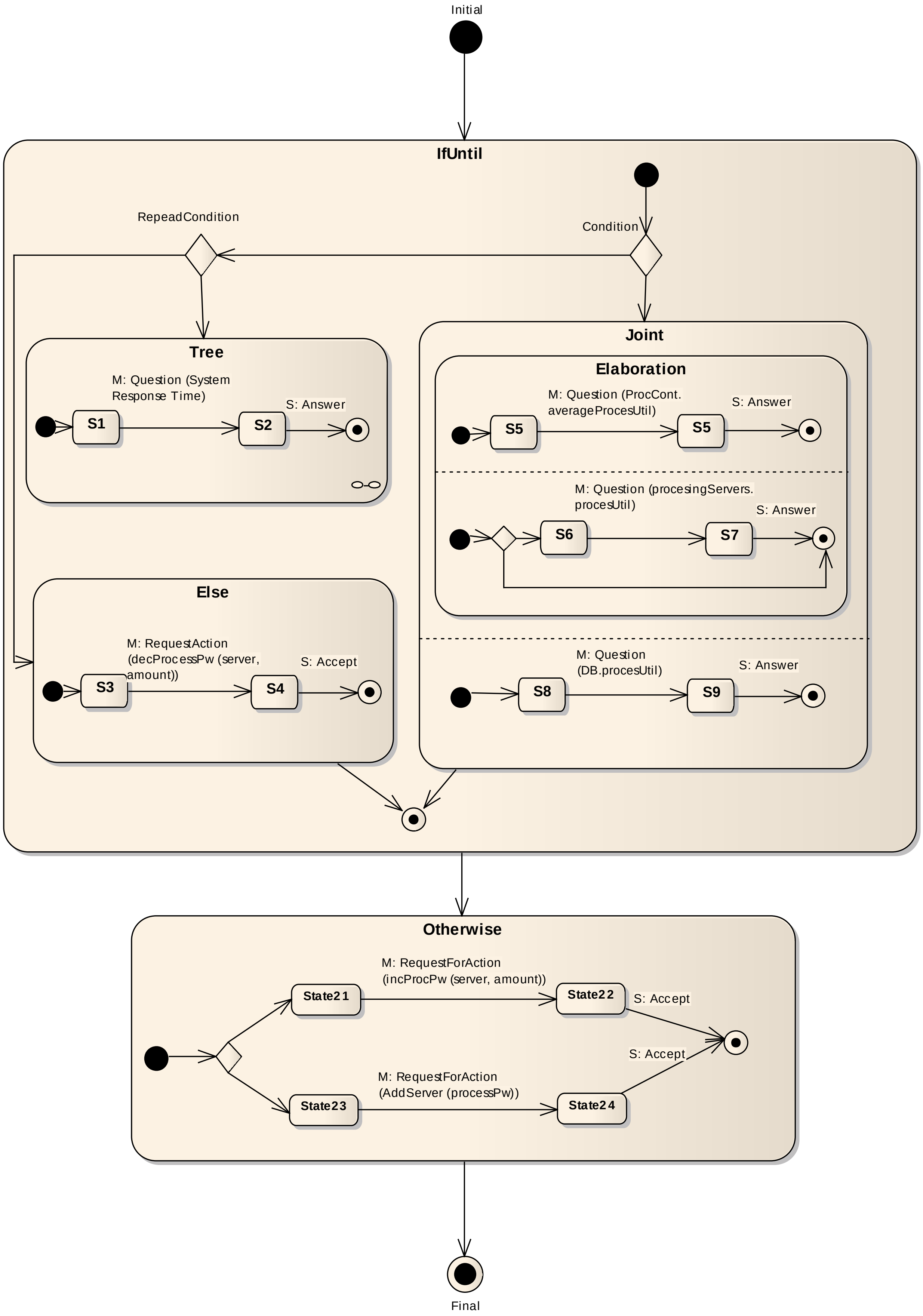


Fig. 11. Complete Discourse Statechart.

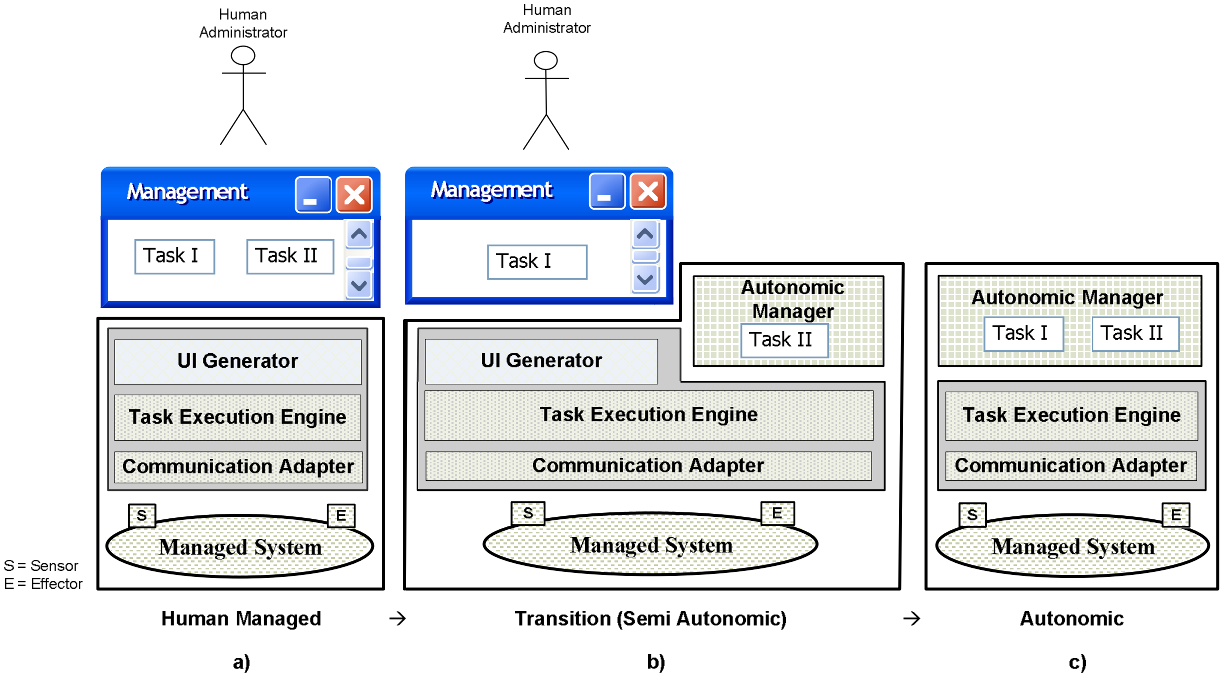


Fig. 12. Autonomic Architecture and Transition Process.

different kinds of servers. It also simulates the workload on such systems. It defines a system as a set of processes in which each process has a set of defined properties and operations. For example, one HTTP server would be represented by one running process. This process would have properties such as CPU power or disk space and would provide operations to change these properties. These processes can also have (incoming or outgoing) connections to others processes. A process can execute the operations on connected processes over these connections. A typical example is the *sendRequest* operation between two servers (processes), which forwards a user request to connected servers.

The simulated system in our case study represents a typical Web store. The simulation tool offers a graphical representation of the simulated system architecture as illustrated in Figure 13. The Web store contains the IT-Infrastructure, which calculated parameters, e.g., response times, dropped requests, etc. Within the IT-Infrastructure, there are three server clusters in charge for distributing HTTP requests, application server requests, and database requests, as common in three-tier architectures. Within each server cluster, servers are responsible for request processing and, if needed, their forwarding. The Shopper process created a sample workload on the system generating *shopper* objects, where each shopper created series of requests to the system. When a request entered the site, the SimSys simulation system time-stamped it for the response time recording and sent it to the HTTP cluster, which forwarded the request to the first ready HTTP server and further via the application cluster to the database. We attached our communication platform to the simulated system via Java method calls.

Let us explain the management discourse shown in Figure 14. Much as in our previous examples, the most significant parameter is the response time (in the SimSys system identified as *latency*). So, the manager can ask a question about the system latency. If the manager considers the latency too large, the manager can ask for the current user activity. As a *result*, the power of the servers can be increased (e.g., if only the system latency has risen) or additional servers can be added (e.g., if both the system latency and the user activity have risen).

In this example, we show a case where a human manager is responsible for the management and communicates with the system using the generated user interface. Figure 15 shows screenshots of this simple management user interface. First, the manager has the possibility to ask a question about the system latency using a button from Screen 1. This represents the *Tree* branch of the *IfUntil* relation. The answer to this question is presented in Screen 2. In addition, it is possible to ask about the user activity (by selecting the button on the right in Screen 2). In this case, the human manager “evaluates‘” the condition of the IfUntil relation. If the manager decides that the latency is too high (in our case 6 seconds), the condition in the IfUntil relation is fulfilled and the manager should be able to continue with the communication (asking the second question by pressing the *getActivity* button). Screen 3 shows the answer to this question and two buttons, *addProcessingPower* and *addServer* representing the possibilities to issue corresponding requests.

This example shows a simple management scenario for the human management case. It demonstrates the feasibility of our approach to handle the required communication (model

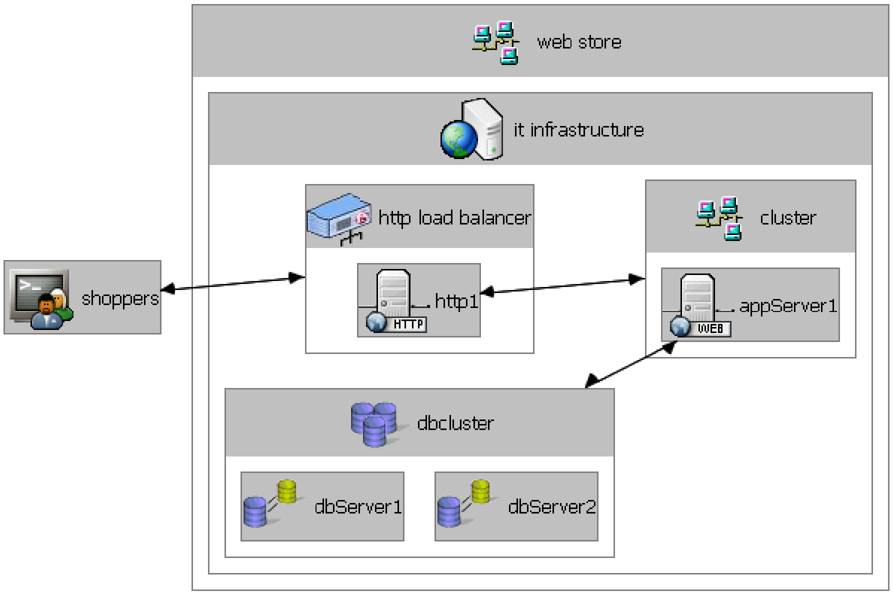


Fig. 13. Simulated System.

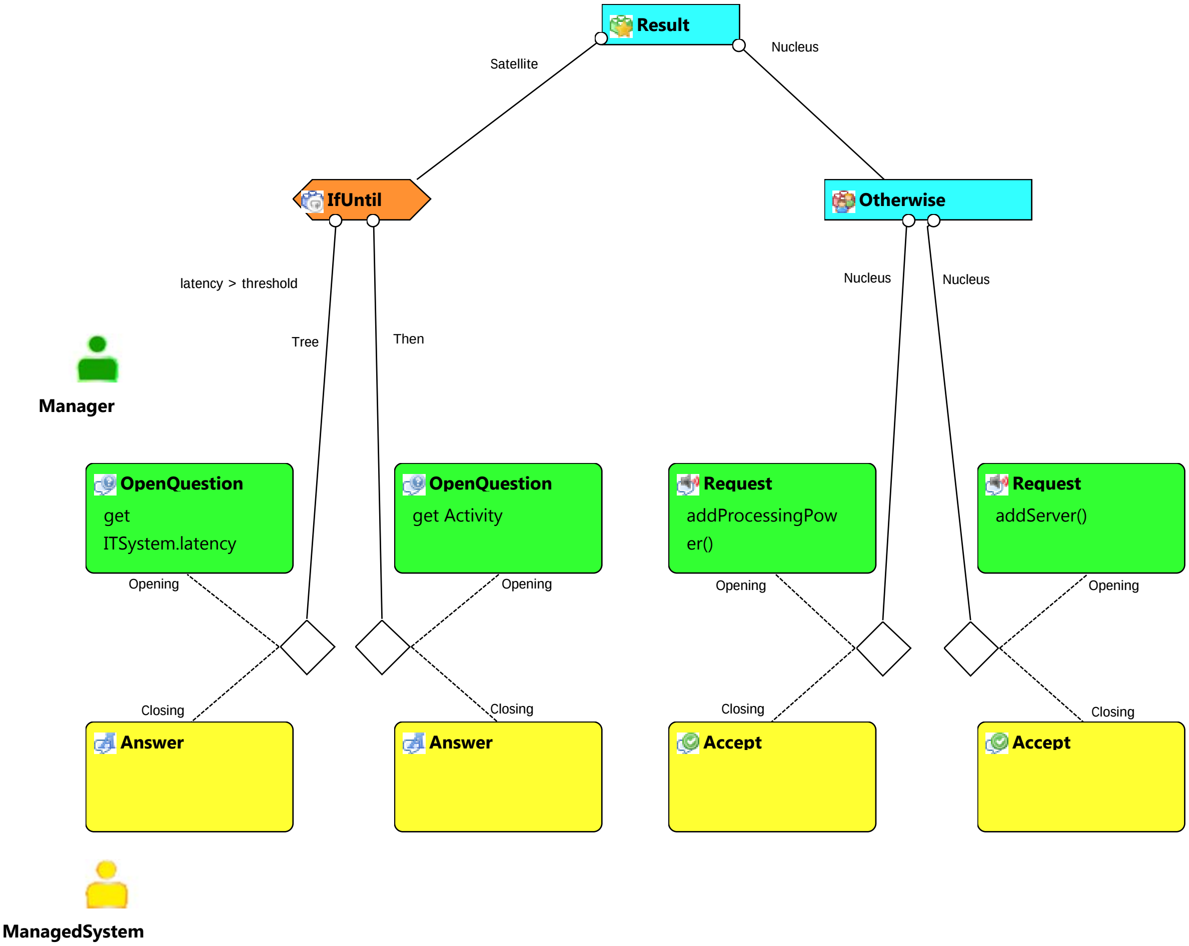


Fig. 14. System Optimizing Discourse Specification.

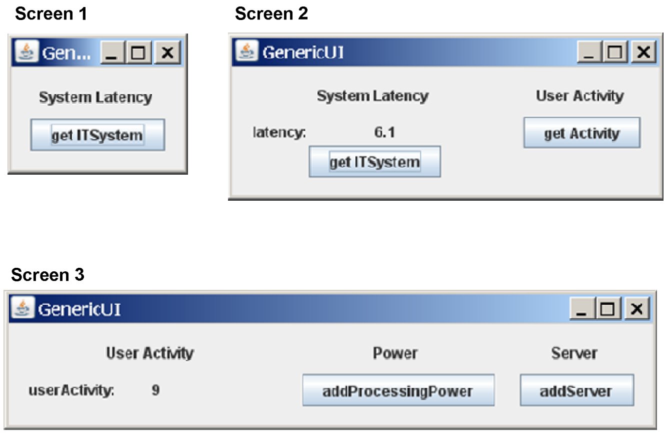


Fig. 15. Screenshots of the Management User Interface.

and execution) between the system to be managed (in this case a simulated three-tier business application) and the human manager.

VIII. Case Study: Application Server JBOSS

This case study demonstrates our approach on the management of a Java application server. In this second case study, we utilized the Java application server JBoss and implemented two management tasks in an autonomic manner.

An application server is a software system which usually takes the role of the "business logic" in the multi-tier architecture. Its main purpose is to encapsulate the access to the data in the database and to simplify data manipulation along the lines of the business process. Usually, an application server utilizes some component model and enables the development of component-based applications. It acts also as a container for these components. An application server offers different features to simplify application development. These include a programming model as a set of APIs, resource handling and pooling, support for distributed computing, authentication and authorization, messaging, transactions, monitoring and control, etc.

In our case study we use the JBoss server, which implements the J2EE industry standard[4] and enables the development of Java components — Enterprise Java Beans (EJBs). For extending JBoss, new components (services) were developed according to the Managed Beans (MBeans) service specification. To be able to perform the case study, especially to be able to monitor required parameters, these were the following extensions:

- a CPU and Memory Monitor, which collects data from the underlying Java Virtual Machine (JVM), and
- a Response Time Recorder. This extension is concerned with gathering information about the response times for client requests.

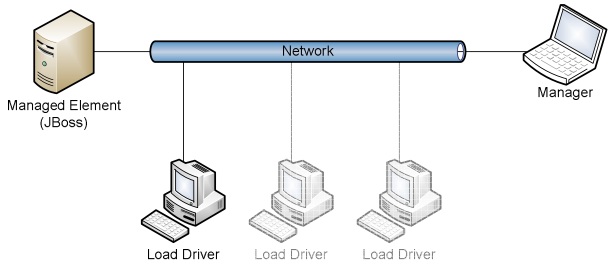


Fig. 16. Case Study Setup.

A. Case Study Setup

The system setup for the case study is shown in Figure 16. It is a distributed setup, with the JBoss application server, the autonomic manager and the load generators installed on different machines. For achieving a more realistic load for the application server, we have used the benchmark tool ECperf [17]. It is designed to measure performance and scalability of J2EE application servers and provides a typical J2EE business application as well as load generators to simulate the workload for such an application.

Since JBoss is a complex software system, which has many parameters that have to be configured, and since the configuration is tedious and error-prone, a self-configuration capability would seem desirable. Therefore, we use the configuration scenario as one example for our case study.

Our second example is about optimization. The optimizing capability is also of interest for a system like JBoss. An optimization in this case would imply the reduction of the response times of client requests sent to the application server. This requires the response times to be measurable and system parameters to be accessible and changeable.

B. Communication Content Model

Prior to creating the management discourses we have to figure out, which parameters are of interest for the monitoring of the JBoss status, as well as to define possible corrective actions which have impact on the behavior of the JBoss server. In essence, we have to define what the manager and the managed server (JBoss) will communicate about.

The most significant JBoss properties for this case study are the following:

- available memory,
- the size of the database connection pool,
- the thread pool size for the EJB invocation processing threads, and
- the server's response time.

The most relevant actions are:

- setting the thread pool size, and
- clearing the EJB cache

Restrictions to the Java Virtual Machine (JVM) heap memory size would also have a deteriorating impact on system performance. Unfortunately, the maximum heap size cannot be changed during runtime, and minimum and maximum values are specified as startup parameters. A modification of these

[4] http://java.sun.com/j2ee/docs.html

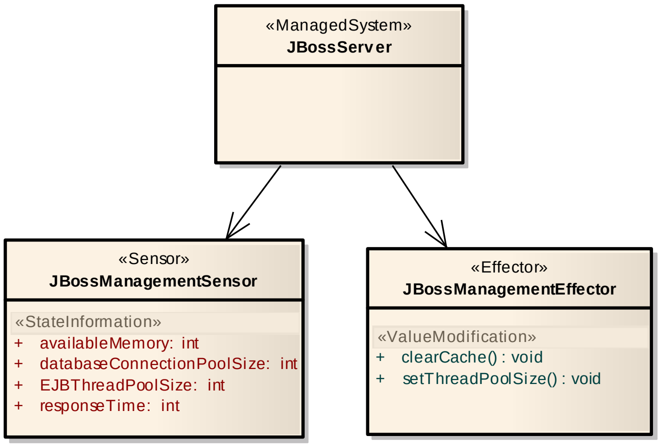


Fig. 17. JBoss Server model for the domain of discourse.

values would thus require a restart of the JVM and the system would be completely out of service for some time, including all applications running within that JVM instance. Therefore, this parameter has not been used in our case study. Also some other parameters have not been included in the case study (SQL statement cache, EJB cache maximum size, etc.).

To use JBoss parameters for the content of communicative acts, we have modeled them as shown in Figure 17. Since we do not model the internal structure of JBoss and are more interested in illustrating our approach to communication, the model is rather simple and contains only one class representing the JBoss server and associated Sensor and Effector interfaces.

C. Configuration Task of the JBoss Server

This example shows the configuration task for the JBoss application server. For a better understanding of this example, let us briefly explain the thread pooling in the JBoss server.

Figure 18 shows the interworking of thread pools. Every request that arrives at JBoss via the Tomcat[5] — embedded server for servlets — has to wait for a thread from the HTTP thread pool to be available in order to be processed. If the client request involves calls to an EJB on the server, the request also has to acquire a thread from the EJB thread pool. And if the EJB call during its execution needs to make a request to the database, a connection from the database connection pool has to be retrieved. The amount of threads and connections in the different pools limits the number of client requests that can be processed concurrently. Having more connections in the database connection pool than threads in the EJB pool is a configuration that is not useful and unnecessarily increases resource usage, because the number of database connections that can be used at the same time is limited by the number of threads in the EJB pool. Thus, it is not desirable to have more connections in the DB connection pool than threads in the EJB thread pool. If the ratio of EJB threads to database connections becomes too small, many client requests will not get a database connection within the configured timeout. For our scenario, we keep the number of database connections constant. The goal is to maintain the number of pooled EJB threads above the number of available database connections. Another goal of this scenario is to prevent an extensive use of memory. The amount of available memory is monitored and when that amount drops below the configured minimum threshold, the memory is freed by:

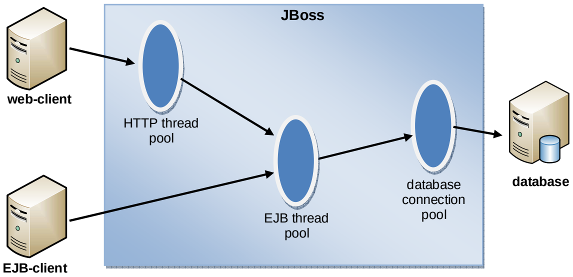


Fig. 18. JBoss threads pooling.

- ordering JBoss to clear up all EJB caches, and by
- reducing the amount of concurrency in request processing by reducing the number of EJB worker threads.

The discourse for this configuration task is shown in Figure 19. Starting from the top of the diagram, the communication for the configuration of the thread pool sizes and the communication for the memory usage managing can be performed in parallel — jointly — as defined by the RST Relation *Joint*. For the configuration of the thread pool sizes the Manager issues *Questions* about the *databaseConnectionPoolSize* and *EJBThreadPoolSize*. If the ratio is not below a given threshold, the manager *requests* the action for setting up the appropriate threadPoolSize. For managing of memory usage, the manager issues the *Question* about the *availableMemory*, and when it drops below some threshold it tries to free it by issuing the *Requests* to *clearCache* and to *reduceThreadPoolSize* of the worker threads.

For the evaluation of the transition towards autonomic systems, an Autonomic Manager has been implemented. It periodically executes the task and discourse in order to perform management functionality. Regarding the thread pooling, it corrects the parameters using the procedure described above. For the memory usage management, we had to limit the maximum amount of memory used by the Java Virtual Machine to 100MB for getting observable effects. As stated previously, the ECperf benchmarking tool was used for load creation. The Autonomic Manager checks the memory status according to the discourse and executes corrective actions whenever needed. Under the same load conditions, JBoss crashed due to *OutOfMemory* exception when the Autonomic Manager has been put out of function.

D. Optimization Task of the JBoss Server

Our second example is the response time optimization task. The most significant parameter having an effect on the response time is EJB thread pool size [18]. The manager asks the *Question* about the current response time and *Requests* the

[5] tomcat.apache.org

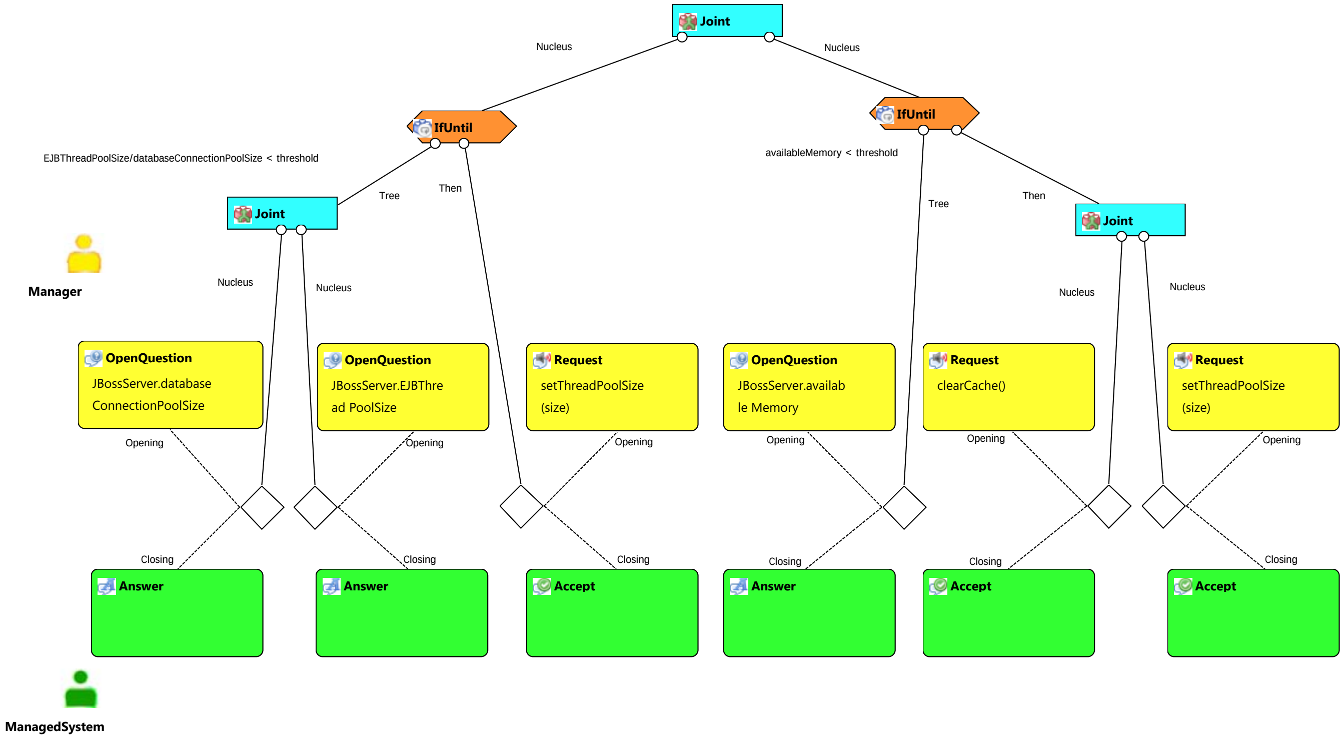


Fig. 19. JBoss configuration discourse.

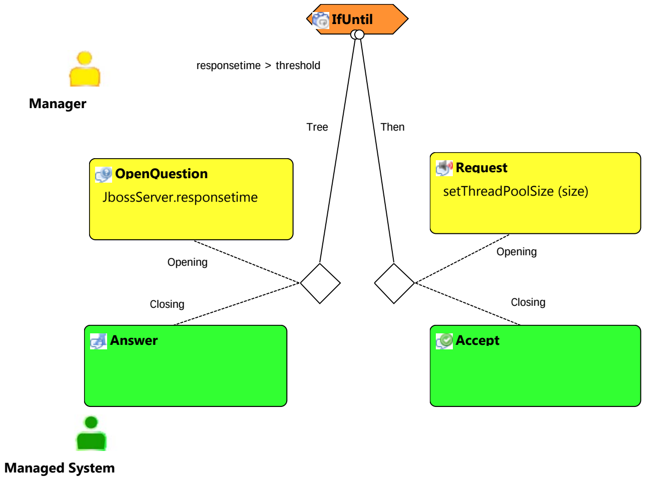


Fig. 20. JBoss optimization discourse.

action for changing the thread pool size, when the response time is below a given limit. Figure 20 shows the corresponding discourse. The communication in this discourse is simple and so are the resulting user interfaces. However, for showing the transition towards autonomic systems, an Autonomic Manager has been implemented as well.

The Autonomic Manager monitors the response time and makes small changes to the thread pool size according to the discourse. Once the response time has improved, it takes the new values of both the thread pool size and response time as a condition for the next execution of the optimization discourse. If and when the modifications have increased the response time, the autonomic manager tries to adjust the parameter in the opposite direction for the next execution of the optimization discourse. We have clearly seen effects of increasing the thread pool size for reducing the response times. However, since more threads use also more memory, this value cannot be increased indefinitely. In our case it settled around 80. We also observed that clearing the EJB caches temporarily increased the CPU load and thus the response times.

IX. Related Work

Our work relates both to the field of interaction modeling (between humans and computers as well as between computers) and to the field of autonomic and self-managed software systems.

Modeling interaction design is mostly done through techniques from task analysis and cognitive science. Techniques based on Hierarchical Task Analysis [19] or GOMS [20] model activities on various levels of detail in a hierarchical way to achieve a particular goal, and (e.g., temporal) relationships between tasks on the same level. On the more detailed levels, task models specify only the type of tasks (e.g., user, system or interaction task) or operators (click, select . . .), but not their intention in the sense of asking, requesting, etc.

Formal interaction modeling is important for interactions between agents. Most approaches for modeling inter-agent communication utilize some form of finite-state machinery. E.g., Labrou and Finin [21] deal with interactions between agents based on KQML, where *conversation policies* are proposed for the description of conversations between agents. Conversation policies represent simple conversations between agents in terms of possible sequences of KQML messages. Our discourse models can represent more complex interactions and should be easier to design by humans.

Management tasks are nowadays performed by well-trained professionals which are responsible for *configuring* the system so that the users can get their jobs done and for *maintaining* the system against both internal failures and internal or external attacks [22]. They interact with the system using command-line interfaces, graphical interfaces or Web-based management tools [23].

Modeling and specifying management tasks and user interfaces for performing those tasks has been neglected in general [24]. However, operators and administrators of software systems are constantly trying to automate administrative tasks and to reduce unnecessary interactions with the system. They use their own executable scripts to automate monitoring of system health, to perform operations on a large number of systems, and to try to eliminate errors on common tasks that take many steps [25]. Our approach also strives for the automation of administrative tasks but concentrates on interactions within such tasks and provides a well-defined way for their modeling.

A typical approach to define automation of software management tasks for autonomic computing is in terms of *policies*. Policies represent instructions to determine the most appropriate activity in a given situation. One way to specify policies has been defined in [26]. They represent policies in the form "`IF condition THEN action`" where *condition* contains a particular state of the system and the *action* represents the actions to be performed if such a state occurs. They also define how to manage and execute such policies. Kephart and Walsh [27] define three types of policies: Action, Goal and Utility Function policies. Action policies are on the lowest level and take also the `if-then` form. Goal policies specify a single desired state and the system should generate behavior itself from the policy. Utility Function policies generalize Goal policies where a desired state is computed by selecting from the present collection of feasible states the one that has the highest utility. The Accord framework [28] defines so-called operational interfaces. This offers the possibility to formulate, inject, and manage rules that are used to manage the runtime behaviors of the autonomic system. Rules incorporate also typical if–then expressions, i.e., "IF condition THEN actions". Similarly to our approach, Cheng et al [29] consider that *"the capturing and representation of human expertise in a form executable by a computer"* is crucial for the automation of management tasks. They have developed a new language for adaptation where the concepts used in the language are derived from system administration tasks. The basic concept in the language is a *tactic*, which embodies a small sequence of actions to fix a specific problem in a localized part of the system. A tactic contains the conditions of applicability, a sequence of actions, and a set of intended effects after execution.

Contrary to this work on policies, our approach focuses on and formalizes interactions between an administrator and a software system. We believe that the interactions are important both for the task execution and for understanding the task. It seems also non-trivial to reduce sometimes complex management tasks to single policies. We believe that our task models should be easier to create by humans since they are based on human communication theories.

X. Discussion

In order to utilize our approach, the system designers and developers would first have to attach our communication platform to the system — to integrate it into the managed system. If the system is designed from scratch, special interfaces would have to be developed. If it is a legacy system, the designers would have to understand and expose the interfaces. This is usually not trivial for such systems. In any case, some additional effort would have to be provided. This may seem to not pay off, especially if the transition towards autonomic operation is far away. However, we believe that our approach can bring some advantages even in the case of (only) human management. By performing task and discourse modeling, the designers become familiar with the system's behavior. Very often, the automation of management tasks is becoming possible only through such an improved understanding.

Our approach adds also some additional overhead to the communication through the need to convert the low-level communication messages into communicative acts. This is more significant for the autonomic management case, where the autonomic manager reacts on the system's events and enacts the corrective actions (possibly after some additional communication with the managed system for problem investigation). Our approach is more directed towards business applications and information systems, where usually the "best-effort" for correcting the problem is sufficient. Anyhow, even this would be much faster than the human reaction. However, due to this overhead, our approach is not well investigated to be used for real-time or embedded systems.

The IT industry has neglected tools and systems used by operators for configuration, monitoring, diagnosis, and repair, and the need for improved user interfaces for operators is large [30]. We believe that the discourse-based communication modeling of administrative (management) tasks, as well as systems modeling for the communication content can contribute to a better understanding of management procedures in general.

XI. Conclusion

In essence, we propose to model software-management interactions and tasks in the form of discourses between the administrator and the software system. In addition to the interaction and task models, we have developed a metamodel for the modeling of the management domain for such tasks. For the execution of these tasks we have defined their procedural semantics, and from these models, we are able to generate user interfaces.

Case studies showed that our approach can be used both in the case of human management as well as in the case of autonomic management. However, since two case studies involved two different managed systems (simulation and real system), management discourses were slightly different. For one managed system and one particular management task, we would have the *same* discourse and therefore the same communication specification both for human and autonomic management. This utilizes to the gradual transition towards autonomic systems.

XII. Acknowledgments

The extension of the JBoss server as well as autonomic managers have been programmed by the diploma student Christoph Bernhard Schwarz, who worked together with the authors of this paper. We also thank Eser Kandogan and IBM for providing us with the simulation tool SimSys.

References

[1] E. Arnautovic, H. Kaindl, J. Falb, and R. Popp, “High-level modeling of software-management interactions and tasks for autonomic computing,” in *Autonomic and Autonomous Systems, 2008. ICAS 2008. Fourth International Conference on*. Washington, DC, USA: IEEE Computer Society, March 2008, pp. 212–218.

[2] E. Arnautovic, H. Kaindl, J. Falb, R. Popp, and A. Szép, “Gradual Transition towards Autonomic Software Systems based on High-level Communication Specification,” in *SAC ’07: Proceedings of the 2007 ACM Symposium on Applied Computing, Autonomic Computing Track*. New York, NY, USA: ACM Press, 2007, pp. 84–89.

[3] J. R. Searle, *Speech Acts: An Essay in the Philosophy of Language*. Cambridge, England: Cambridge University Press, 1969.

[4] M. Nowostawski, D. Carter, S. Cranefield, and M. Purvis, “Communicative acts and interaction protocols in a distributed information system,” in *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS’03)*. New York, NY, USA: ACM Press, 2003, pp. 1082–1083.

[5] J. Falb, R. Popp, T. Röck, H. Jelinek, E. Arnautovic, and H. Kaindl, “Using communicative acts in high-level specifications of user interfaces for their automated synthesis,” in *Proceedings of the 20th IEEE/ACM International Conference on Automated Software Engineering (ASE’05)*. New York, NY, USA: ACM Press, 2005, pp. 429–430, tool demo paper.

[6] P. Luff, D. Frohlich, and N. Gilbert, *Computers and Conversation*. London, UK: Academic Press, January 1990.

[7] W. C. Mann and S. Thompson, “Rhetorical Structure Theory: Toward a functional theory of text organization,” *Text*, vol. 8, no. 3, pp. 243–281, 1988.

[8] B. Urgaonkar, P. Shenoy, A. Chandra, P. Goyal, and T. Wood, “Agile dynamic provisioning of multi-tier internet applications,” *ACM Trans. Auton. Adapt. Syst.*, vol. 3, pp. 1–39, 2008.

[9] J. Falb, H. Kaindl, H. Horacek, C. Bogdan, R. Popp, and E. Arnautovic, “A discourse model for interaction design based on theories of human communication,” in *CHI ’06 Extended Abstracts on Human Factors in Computing Systems*. New York, NY, USA: ACM Press, 2006, pp. 754–759.

[10] C. Bogdan, J. Falb, H. Kaindl, S. Kavaldjian, R. Popp, H. Horacek, E. Arnautovic, and A. Szep, “Generating an abstract user interface from a discourse model inspired by human communication,” in *Proceedings of the 41th Annual Hawaii International Conference on System Sciences (HICSS-41)*. Piscataway, NJ, USA: IEEE Computer Society Press, January 2008.

[11] R. Popp, J. Falb, E. Arnautovic, H. Kaindl, S. Kavaldjian, D. Ertl, H. Horacek, and C. Bogdan, “Automatic generation of the behavior of a user interface from a high-level discourse model,” in *Proceedings of the 42nd Annual Hawaii International Conference on System Sciences (HICSS-42)*. Piscataway, NJ, USA: IEEE Computer Society Press, 2009.

[12] J. Falb, R. Popp, T. Röck, H. Jelinek, E. Arnautovic, and H. Kaindl, “Fully-automatic generation of user interfaces for multiple devices from a high-level model based on communicative acts,” in *Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS-40)*. Piscataway, NJ, USA: IEEE Computer Society Press, Jan 2007.

[13] I. Horrocks, *Constructing the User Interface with Statecharts*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1999.

[14] M. McKinlay and Z. Tari, “Dynwes - a dynamic and interoperable protocol for web services,” 2002, pp. 74–83.

[15] *An architectural blueprint for autonomic computing*, 3rd ed., IBM Corporation, June 2005, white Paper.

[16] E. Kandogan, C. Campbell, P. Khooshabeh, J. Bailey, and P. Maglio, “Policy-based management of an e-commerce business simulation: An experimental study,” *Autonomic Computing, 2006. ICAC ’06. IEEE International Conference on*, pp. 33–42, 13-16 June 2006.

[17] S. M. Inc., “Ecperf (tm) specification,” Sun Microsystems Inc., 2002.

[18] Y. Zhang, W. Qu, and A. Liu, “Adaptive self-configuration architecture for j2ee-based middleware systems,” *System Sciences, 2006. HICSS ’06. Proceedings of the 39th Annual Hawaii International Conference on*, vol. 9, pp. 213a–213a, Jan. 2006.

[19] Q. Limbourg and J. Vanderdonckt, “Comparing task models for user interface design,” in *The Handbook of Task Analysis for Human-Computer Interaction*, D. Diaper and N. Stanton, Eds. Mahwah, NJ, USA: Lawrence Erlbaum Associates, 2003, ch. 6.

[20] B. E. John and D. E. Kieras, “Using GOMS for user interface design and evaluation: Which technique?” *ACM Trans. Comput.-Hum. Interact.*, vol. 3, no. 4, pp. 287–319, 1996.

[21] Y. Labrou and T. Finin, “Semantics and conversations for an agent communication language,” in *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI’97)*, 1997, pp. 584–591.

[22] E. A. Anderson, “Researching system administration,” Ph.D. dissertation, University of California, Berkeley, 2002.

[23] R. Barrett, E. Kandogan, P. P. Maglio, E. M. Haber, L. A. Takayama, and M. Prabaker, “Field studies of computer system administrators: analysis of system management tools and practices,” in *CSCW ’04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*. New York, NY, USA: ACM Press, 2004, pp. 388–395.

[24] R. Barrett, Y.-Y. M. Chen, and P. P. Maglio, “System administrators are users, too: designing workspaces for managing internet-scale systems,” in *CHI ’03: CHI ’03 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM Press, 2003, pp. 1068–1069.

[25] E. Kandogan and J. Bailey, “Usable Autonomic Computing Systems: The Administrator’s Perspective,” in *ICAC ’04: Proceedings of the First International Conference on Autonomic Computing (ICAC’04)*. Washington, DC, USA: IEEE Computer Society, 2004, pp. 18–26.

[26] R. M. Bahati, M. A. Bauer, and E. M. Vieira, “Mapping Policies into Autonomic Management Actions,” in *ICAS ’06: Proceedings of the International Conference on Autonomic and Autonomous Systems*. Washington, DC, USA: IEEE Computer Society, 2006, p. 38.

[27] J. O. Kephart and W. E. Walsh, “An Artificial Intelligence Perspective on Autonomic Computing Policies,” in *POLICY ’04: Proceedings of the Fifth IEEE International Workshop on Policies for Distributed Systems and Networks*. Washington, DC, USA: IEEE Computer Society, 2004, p. 3.

[28] H. Liu and M. Parashar, “Accord: A Programming Framework for Autonomic Applications,” *Systems, Man and Cybernetics, Part C, IEEE Transactions on*, vol. 36, no. 3, pp. 341–352, May 2006.

[29] S.-W. Cheng, D. Garlan, and B. Schmerl, “Architecture-based Self-adaptation in the Presence of Multiple Objectives,” in *ICSE 2006 Workshop on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, Shanghai, China, 21-22 May 2006.

[30] D. Oppenheimer, A. Ganapathi, and D. A. Patterson, “Why do internet services fail, and what can be done about it?” in *USITS’03: Proceedings of the 4th conference on USENIX Symposium on Internet Technologies and Systems*. Berkeley, CA, USA: USENIX Association, 2003, pp. 1–1.

Perception of Utility in Autonomic VoIP Systems

Edward Stehle, Maxim Shevertalov, Paul deGrandis, Spiros Mancoridis, Moshe Kam
Department of Computer Science
Drexel University
Philadelphia, PA 19104, USA
Email: {edward, max, pd442, spiros} @drexel.edu and kam@minerva.ece.drexel.edu

Abstract

The transmission of voice-over-Internet protocol (VoIP) network traffic is used in an increasing variety of applications and settings. Many of these applications involve communications where VoIP systems are deployed under unpredictable conditions with poor network support. These conditions make it difficult for users to configure and optimize VoIP systems and this creates a need for self configuring and self optimizing systems. To build an autonomic system for VoIP communications, it is valuable to be able to measure the user perceived utility of a system. In this paper we identify factors important to the estimation of user perceived utility in task dependent VoIP communications.

Keywords-autonomic; VoIP; utility function;

1. Introduction

As the transmission of voice-over-Internet protocol (VoIP) network traffic becomes commonplace, VoIP is used in an increasing variety of applications and settings. Many current applications are outside the context of simple social conversation across dependable networks. Field applications, such as military operations, employ VoIP for task-specific communications and require VoIP to operate under poor network conditions. Emergency-response personnel may use VoIP communications to complete tasks in disaster areas where extreme weather or other adverse conditions interfere with network performance. Operations may be carried out in locations where there is little or no communications infrastructure or where the communications infrastructure has been damaged. Under these field conditions VoIP needs to be served by small, mobile, ad-hoc networks with limited resources.

VoIP systems for field communications need to be deployed quickly to minimize response time. In order to deliver the best possible support to field operations, VoIP systems must be optimized to the field conditions. This creates a difficult problem for the users of field VoIP systems. How do you quickly find an optimal configuration for a VoIP network under adverse conditions when little is known about these conditions before the system arrives in the field? How do you optimally manage a VoIP network under changing field conditions? This is an ideal application for autonomic systems. If we can produce a context aware VoIP system that can self configure when deployed and self optimize as field conditions change, we can reduce deployment time and improve overall performance in unknown and unpredictable settings.

In order to build an autonomic system for field VoIP communications, we must have a way to measure the performance of the system. Such an autonomic system must be aware of user perceived utility of the VoIP application. One approach when including "black-box" applications in an autonomic system, is to develop models for application utility estimation [2]. Autonomic systems using utility function policies [3], [4] require an estimate of an application's performance. Previous work in the area of monitoring the health of autonomic systems involved the use of a pulse to estimate the health of specific autonomic elements [5], [6], [7].

In this paper we look at methods to map network conditions to user-perceived utility as a utility function. We will review the findings of our earlier work [1]. We identify factors that need to be considered when mapping network conditions to user perceived utility. Specifically, we determine if the mapping from network conditions to perceived utility is task dependent. We also determine if the mappings for users performing different roles within the same task are affected by their roles. We will compare the results of our previous experiments [1] with the E-Model. Finally, we wish to determine if perceived utility changes with the continued repetition of a task.

This paper is structured as follows. First we present previous work in calculating the user perceived utility of VoIP applications (Section 2). Then we will present the set up of our human subject experiments to explicitly determine user perceived utility of VoIP applications (Section 3). We will conclude by presenting our results (Section 4), concluding remarks (Section 5), and an appendix of collected data (Section 6).

2. Previous Work

Existing approaches for predicting user perception of utility in VoIP systems fall into two main categories. Some approaches base predictions on the degradation of a reference signal and other approaches map network conditions to perception of utility based on subjective data gathered in human-subjects testing.

2.1. Reference Signal Approach

Objective systems such as the Perceptual Speech Quality Measure (PSQM) [14] and the Perceptual Assessment of Speech Quality (PESQ) [13] require a speech sample to be sent across a VoIP network. The original sample is then compared to the sample that is received on the other end of the VoIP system. A prediction of user utility is made based on the degree to which the signal has degraded.

The main criticism of the existing objective approaches is that they only consider signal distortion in one direction. They do not consider network impairments such as delay and echo [12].

2.2. Subjective Testing based Approach (E-Model)

The most common model for mapping network conditions to user-perceived utility for voice applications is the E-model [10]. During the mid-nineteen nineties the International Telecommunications Union (ITU) designed the E-Model to measure objectively the quality of a public-switched telephony network (PSTN). The E-Model was originally intended to be used by network planners to predict the quality of PSTNs without the need for expensive and time-consuming testing of human subjects. It has since been adapted to cellular communications and IP telephony [11], [9], [8].

The E-Model uses a transmission rating factor as a measure of the predicted network quality. The transmission rating factor R is the linear sum of various Impairment factors and an expectation compensation factor. This linear sum is described in Equation 1.

$$R = Ro - Is - Id - Ie + A \qquad (1)$$

The first variable Ro is the baseline of the model for the given network. This is the E-Model value of the unimpaired network. The most commonly used baseline value for an ideal unimpaired network is one hundred. If the network does not perform ideally in the absence of impairment factors it may be given a lower baseline value. The Is impairment factor is defined as simultaneous impairment, which is the sum of impairments occurring simultaneous to voice. This includes impairments such as inappropriate volume and sidetone, which cannot be separated from voice. Sidetone is any sound from the earpiece of a phone that is picked up by the mouthpiece of the phone. The primary effect of sidetone is echo. Most studies that use the E-Model for evaluating VoIP calls do not include simultaneous impairments since they are ”intrinsic to the voice signal itself and do not depend on the transmission over the network” [8]. The Id impairment factor is the impairment caused by the round trip delay of the voice signal. Any Impairment caused by the use of specific equipment is included in the Ie factor. This factor includes distortion of the original signal due to the codec, the packet loss in the network and the packet loss in the playback buffer. The final factor A serves as a method to compensate for the expectation or other advantages derived from using IP telephony. For instance, most people expect that over traditional telephone wire the call would be very good but are a little more forgiving when speaking over a mobile phone.

The E-Model has become a commonly used metric to predict the quality of VoIP applications for several reasons. Most models for objective quality measurement require that the received signal be compared to the sent signal. The E-Model is the only widely recognized metric that does not require a reference signal, making it computationally feasible for real time applications. In addition, the E-Model correlates well with subjective quality in situations where IP telephony functions in the same fashion as PSTN; for example in local VoIP networks where anomalous traffic conditions are minimized.

The E-Model transmission rating factor R can be mapped to a Mean Opinion Score (MOS) by the use of a function described by Cole and Rosenbluth [11]. MOS is a scoring system commonly used in tests involving human subjects. Subjects using MOS rate the quality of a VoIP system with a score from one to five where one is the worst quality and five is the best. The function for converting R is illustrated in Equation 2, Table 1 and Figure 1.

$$f(r) = \begin{cases} 1 & r \leq 0 \\ 1 - \frac{7}{10^3} r + \frac{7}{6250} r^2 + \frac{7}{10^6} r^3 & 0 < r < 100 \\ 4.5 & r \geq 100 \end{cases} \quad (2)$$

Table 1. Mapping of Transmission Rating Factor to the Mean Opinion Scale

Transmission Rating Factor Scale of 0-100	**User Satisfaction**	**Mean Opinion Score Scale of 1-5**
90-100	Best	4.34-4.5
80-90	High	4.03-4.34
70-80	Medium	3.60-4.03
60-70	Low	3.14-3.60
50-60	Poor	2.58-3.1
0-50	Worst	4.34-4.5

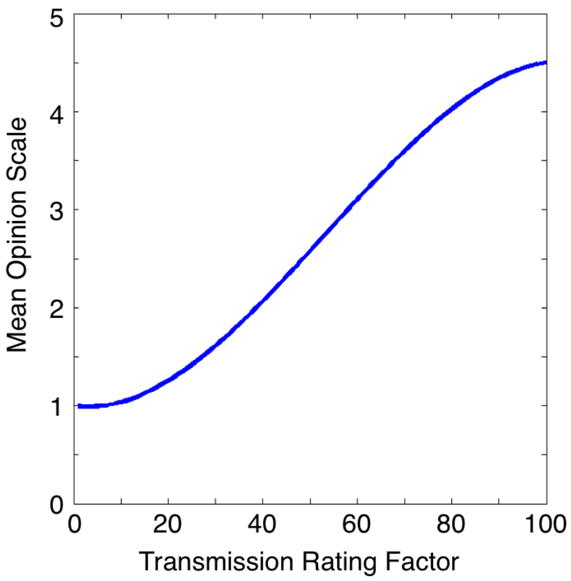


Figure 1. Mapping of Transmission Rating Factor to the Mean Opinion Scale

There are however problems with using the E-Model to predict user satisfaction with VoIP. Although the E-Model correlates well with subjective quality in situations where IP telephony functions in the same fashion as PSTN, using the E-Model in the context of the Internet greatly decreases such correlation. The E-Model was not derived for this explicit purpose. In fact the E-Model was not intended as a quality assessment tool, but rather tool for planning circuit switched networks. The E-Model was not meant to be applied to IP networks. The impairment factors that comprise it deal more with signal processing than with IP networks. The E-Model does not consider the differing expectations users may have toward delay when using VoIP over the Internet. Delay in IP networks is greater than delay in PSTNs. When using a large congested and unpredictable IP network such as the Internet the delay can be much greater than in PSTNs. Users who are used to dealing with delays when using Internet applications may be more tolerant of delay when using VoIP over the Internet. Although the E-Model includes a variable to compensate for user expectations it is independent of the impairment factors. Internet VoIP users may be more tolerant of delay, but not more tolerant of loss. This cannot be captured by the expectation compensation factor A in E-Model. The compensation factor adds a constant independent of the impairment factors.

2.3. Problems with current approaches

Neither reference-signal based approaches nor subjective test approaches consider the impact of task on a perceived utility. Current models assume that, given network conditions, users will always perceive utility in the same manner regardless of what task they are using VoIP to perform. In tests using circuit-switched networks Kitawaki and Itoh concluded that speech quality due to propagation delay greatly depends on the kind of task [12]. Their tests showed that delay has a greater effect on tasks that are more interactive.

3. Our Tests

In our tests, subjects rate the quality of VoIP under varying network conditions. Each test involves one pair of human test subjects. The subjects carry out a series of similar tasks that require communication using a VoIP application. For all of our testing we used Gnome Meeting as the VoIP application and G.711 for our audio codec. We vary the network conditions using a FreeBSD application named Dummynet, which allows us to set the bandwidth, latency and loss of the link used by our test subjects. A single test point in our experiment is a 3-tuple (bandwidth, latency, loss). Each of these parameters can have one of five values. We test across all combinations of these values, giving

Table 2. 3-tuple Parameters

Parameter	Values
Bandwidth	25, 40, 50, 65, 80 (kbps)
Latency	0,1000, 2000, 3000, 4000 (ms)
Loss	0, 12.5, 25, 50, 60 (percent)

us 125 points per subject. The possible values of the parameters are listed in Table 2.

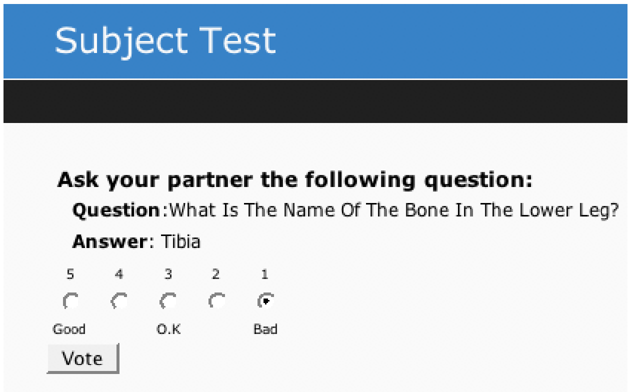
We have been performing three different types of human subject tests, each with a different task. We believe that the relationship between network conditions and user satisfaction is task dependent and that using more than one test with different tasks will provide data to support this belief. All of the tests have the same basic structure. There are two roles that the subjects play during a test. One subject is a `questioner` and one subject is a `responder`. The actual duties of the `questioner` and the `responder` vary between the types of test. The subjects perform one task at each of the 125 test points. After a task is completed each subject votes on the quality of the communication. Then the network conditions are changed to the next point and the next task begins. The subjects rate the quality on a scale of one to five where one is bad, five is good, and three is okay. The subjects alternate between the roles of questioner and responder after each task. Each test collects 250 data points and takes between 60 and 90 minutes to complete.

3.1. Simple Information Exchange Test

The first VoIP test is designed to measure perceived utility during tasks involving a simple exchange of information. The tasks in this test involve the `questioner` asking a trivia question and the `responder` answering it. Completion of this task involves minimum back-and-forth conversation between the subjects and does not have any time constraint. We believe that this test is useful for modeling VoIP communications where the users are simply exchanging facts or instructions. For example, if VoIP is being used to convey a military target's position and instructions for engaging the target, we expect the conversation to be limited to conveying position, conveying instructions, and a confirmation that the message has been received.

In this test the `questioner` is given a trivia question and the answer to the trivia question. A screenshot of our testing application with a sample question can be seen in Figure 2. The `responder` is given a list of possible answers, one of which is correct. A screenshot of the responders answers can be seen in Figure 3. The `questioner` reads the question to the `responder`. The `responder` picks an answer from the list and reads it to the `questioner`. Then the `questioner` records whether the question was answered correctly. This requires both subjects to receive a piece of information from the other and then respond to that information.

We have conducted the simple information test with thirty human subjects and collected 3750 data points.

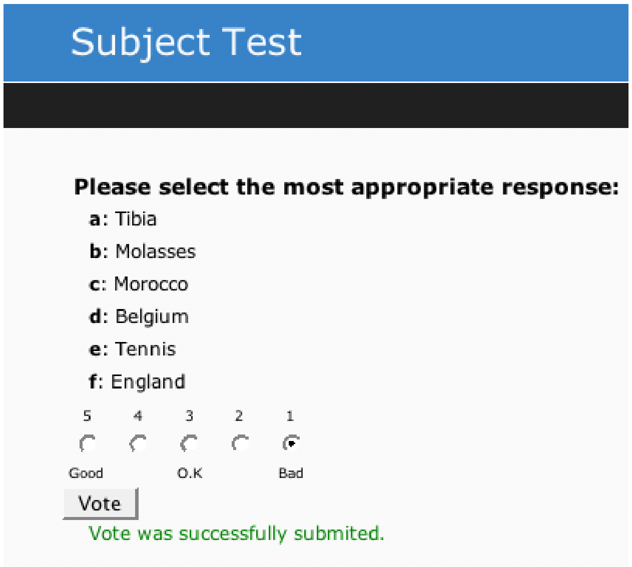


Ask your partner the following question:

Question:What Is The Name Of The Bone In The Lower Leg?

Answer: Tibia

Figure 2. Simple Information Exchange Test Questioner Screen



Please select the most appropriate response:

a: Tibia

b: Molasses

c: Morocco

d: Belgium

e: Tennis

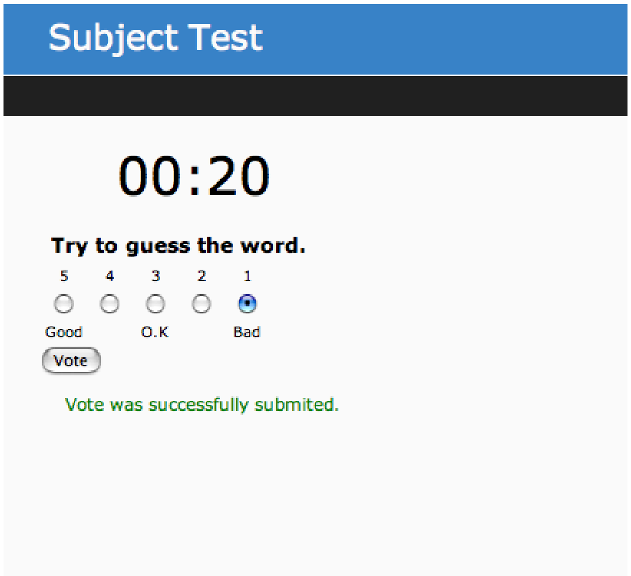
f: England

Vote was successfully submited.

Figure 3. Simple Information Exchange Test Responder Screen

3.2. Time-Sensitive Collaboration Test

The second test is designed to measure perceived utility during time-sensitive tasks that involve some collaboration between subjects. The tasks in this test involve a considerable amount of back-and-forth conversation between the two subjects in order complete a time-constrained task. This test is intended to model situations where users are not trying simply to convey information but to perform some collaborative task. For example, if two military commanders need to collaborate on a plan for a time-critical task, we would expect a considerable amount of back-and-forth conversation and pressure to complete the plan quickly.



Subject Test

00:20

Try to guess the word.

○ ○ ○ ○ ●

Vote was successfully submited.

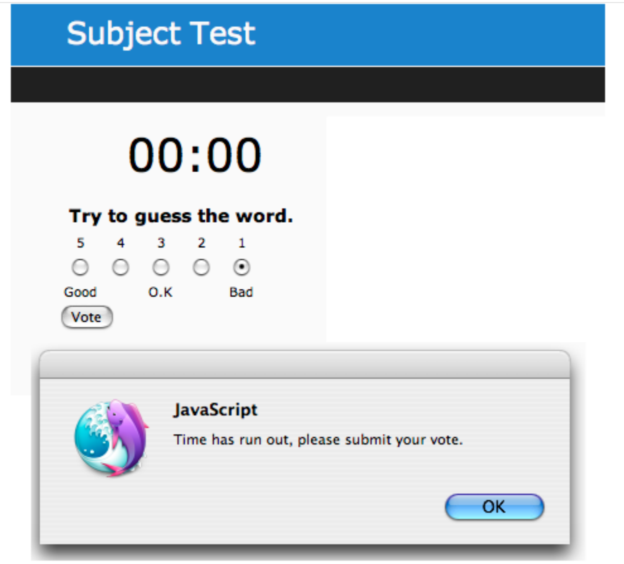
Figure 4. Time-Sensitive Collaboration Test Responder Screen

In this test the questioner is given a word that the responder must correctly guess, but the questioner may not explicitly state the word. Screenshots of the Time-Sensitive Collaboration Test can be seen in Figures 4 and 5. The questioner can only describe the word and answer the questions of the responder. The responder can guess the word or ask the questioner for specific information about the word. Each task has a time limit of thirty seconds. The task ends when the responder correctly guesses the word or the time runs out.

We have conducted the time-sensitive collaboration test with 30 human subjects and collected 3750 data points.

3.3. Time-Sensitive Information Exchange

The third VoIP test is designed to measure perceived utility during time constrained tasks involving the exchange of multiple pieces of information. The tasks in this test involve the collaborative summing of a series of small integers within a limited period of time. This test is intended to model situations where users need to collaborate and the collaboration is limited to a series of simple exchanges of information. For example, in order to coordinate the response of emergency workers in separate locations of a disaster area these workers may need to combine collected data such as the number of disaster victims.



Subject Test

00:00

Try to guess the word.

○ ○ ○ ○ ●

JavaScript

Time has run out, please submit your vote.

Figure 5. Time-Sensitive Collaboration Test Responder Screen Time Expired

In this test the questioner and responder are each given a list of integers. The questioner is given a “starting number”, an “ending number” and two “adding numbers”. The responder is given three “adding numbers”. The starting number is an integer from zero to ten, the adding numbers are integers from zero to five, and the ending number is the sum of the starting number and the adding numbers. The questioner initiates the task by reading the starting number to the responder. The responder adds his first adding number to the starting number and reads the sum to the questioner. The exchange continues with each subject adding one adding number to the sum until all of the adding numbers have been summed with the starting number. Once all of the adding numbers have been summed with the starting number the questioner checks the total against the ending number and informs the responder that the numbers have been summed correctly or incorrectly. Each task has a time limit of thirty seconds.

We have conducted the time sensitive collaboration test with 30 human subjects and collected 3750 data points.

3.4. User-Adjustment Tests

User-adjustment tests were designed to measure changes in perceived utility as a task is repeated. The tasks in these tests are performed over a set of network conditions, and then repeated over the same set of network conditions. The results from the first time through the set of network conditions can then be compared to the results from the second time through the same set of network conditions. These tests are designed to model situations where a user learns and adjusts to tasks.

User adjustment tests were performed using the three previously described tasks. These include the tasks described in Section 3.1 (Simple Information Exchange Test), Section 3.2 (Time Sensitive Collaboration Test), and Section 3.3 (Time Sensitive Information Exchange). In their original form, each of the previously described tests was performed over 125 network condition points. Repeating all points in a test would yield a test with 250 data points that would take two to three hours to complete. A test of this length would tire the test subject. This would corrupt the test results and create unnecessary stress for the test subjects. In order to reduce the time required to complete test trials the size of the set of network settings was reduced. The possible values of network condition parameters described in Table 3 were altered so that only the highest bandwidth value was used. A single test point in our user adjustment tests is a 2-tuple (latency, loss). Each of these parameters can have one of 5 values, giving us 25 points. These points are randomly ordered, and then repeated in the same random order, giving us 50 points per subject. The set of possible parameters for the user adjustment tests is listed in Table 3.

Table 3. 2-tuple Parameters

Parameter	Values
Latency	0,1000, 2000, 3000, 4000 (ms)
Loss	0, 12.5, 25, 50, 60 (percent)

3.5. Our Test Bed

In order to carry out these tests we created a test bed that allows two subjects to converse using VoIP while we control the properties of the channel over which VoIP is running.

Our test bed consists of one “subject computer” for each of our two subjects, a switch partitioned into two subnets, and one “bridge computer” that is used to set the bandwidth, latency and loss of the channel over which the two subject computers communicate. Figure 6 illustrates the manner in which the test bed is connected. Each of the subject computers is connected to a different subnet and the bridge computer is connected to both of the subnets. Communications between the two subject computers are routed through the bridge computer. The bridge computer employs Dummynet to enforce the bandwidth, latency and loss on the channel connecting the two subject computers. The subject computers and the bridge computer are also connected through a back channel, which is not effected by Dummynet. This back channel is used to send messages to the bridge computer instructing it to change the Dummynet settings.

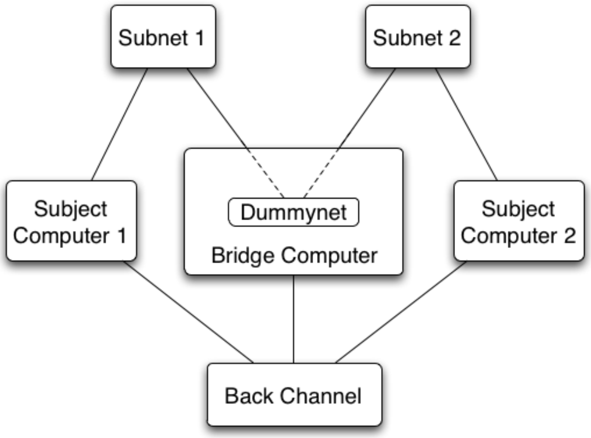


Figure 6. Architecture of the Test Bed

4. Results

The results of our experiments can be seen in Figures 7-12 found in the Appendix (Section 6). There are two types of figures: tests in which the test points are 3-tuples (bandwidth, latency, loss) that are represented by three-dimensional plots and tests in which the test points are 2-tuples (latency, loss) that are represented by two-dimensional plots.

The three-dimensional plots show the space defined by bandwidth, loss and latency measurements. Within this space color is used to represent a user-satisfaction rating. The darkest red represents the areas that were rated best, and the darkest blue represents the areas that were rated worst. In each of these figures our test space is represented by three plots, each sliced along a different axis. One is cut along bandwidth, one along latency, and one along loss.

The two-dimensional plots show the space defined by our loss and latency measurements. The same color convention is used to represent user satisfaction rating.

4.1. Different Tasks

In this section we present the results of our Simple Information Exchange Test, Time-Sensitive Information Exchange and Time-Sensitive Collaboration Test. Descriptions of each of these tests can be found in Section 3 and the results can be seen in Figures 7, 8 and 9. The average variance, minimum variance, maximum variance and the variance of the variance for all test points is shown in tables 4 through 6.

As expected, the results vary somewhat for different tasks. One obvious difference between the results for different tasks is the effect of latency on utility. In the time-sensitive collaboration test and in the time-sensitive information exchange test, latency had a greater effect on perceived utility than in the simple information-exchange test. These results make intuitive sense. Tests in which the tasks are subject to time constraints show a greater user reaction to latency. We believe that this is caused not only by the addition of the time constraints, but also by the collaborative nature of the communication. During this type of collaboration, subjects spend more time speaking back-and-forth than they do during the simple information exchange test. Greater latency can cause this back-and-forth communication to fall out of sync, creating additional difficulties in communication.

Another obvious difference is the effect of bandwidth and loss. Bandwidth has the greatest effect on the simple information-exchange test. We believe that the collaborative nature of the time-sensitive tests helps users adjust to poor voice quality. Because these tests involve more back-and-forth communication, the users have more opportunity to recognize poor quality. Once poor voice quality is recognized, users may begin to employ strategies such as repeating messages without being asked. The back-and-forth communication also gives users more opportunity to recognize conversational context. Recognizing conversational context can be helpful for filling in portions of messages which cannot be understood.

Table 4. Variance of User Perceived Utility for Simple Information Exchange

Average Variance	0.732
Maximum Variance	2.193
Minimum Variance	0.216
Variance of Variance	0.080

Table 5. Variance of User Perceived Utility for Time Sensitive Collaboration

Average Variance	0.610
Maximum Variance	1.140
Minimum Variance	0.127
Variance of Variance	0.040

Table 6. Variance of User Perceived Utility for Time Sensitive Information Exchange

Average Variance	0.740
Maximum Variance	2.187
Minimum Variance	0.187
Variance of Variance	0.101

4.2. Different Roles in a Task

Within each of the tasks described in Section 3 one test subject act as a questioner and one test subject acts as a responder. Figure 10 shows the results of the Time Sensitive Information Exchange test for both responder and questioner, responder only, and questioner only.

When the results of our test are split into questioner-only and responder-only plots it is clear that the role played within a task has an effect on perceived utility. Again, this is an expected result. Different roles within a single test can be thought of as different sub-tasks, and we have already illustrated that perceived utility is task dependent.

4.3. User Adjustment

In the User Adjustment tests described in Section 3.4, we have subjects carry out tasks over the same test points two times in a row. The purpose of tests if to determine if the test subjects adjust to adverse network conditions while performing tasks. The results of subjects performing the Simple Information Exchange test over the same set of 25 points two times in a row can be found in Figure 11.

The results of our user adjustment tests show perceived utility changes as users repeat a task. In each of the tests the variance of the perceived utility decreased during the second time through the test points. At the same time the average perceived utility stayed approximately the same. It appears that as users repeat a task over different network conditions they “get used to it”. They perceive fewer extremes in utility and tend to perceive a larger portion of the test space as “okay”.

4.4. Comparison to E-model

The E-model equation for predicting user perception of utility is described in Section 2.2. It is the most commonly used tool for prediction of user utility in VoIP systems. Figure 12 shows a comparison of the E-model to our test results for the Simple Information Exchange test.

Our results differ greatly from the predictions of he E-model. We believe that this difference is due to task oriented nature of our tests. The E-model was created to predict user perceived utility in circuit switched phone systems. These phone systems are designed to handle not only task oriented communications, but also social conversations. We believe a user given a task to complete is less likely to dismiss a communication session due to impaired quality than a user having a social conversation. The user attempting to complete a task is more likely to fine utility in an impaired connection that allows them to complete their task.

5. Summary and Conclusion

Knowledge of network conditions, such as bandwidth, latency and loss, is not sufficient to predict the performance of a VoIP system adequately. The predictor must also have knowledge of the task being performed over the VoIP system. Our tests show that user perceived utility may be very different for users performing different tasks even if network conditions are the same.

Many tasks performed over VoIP systems involve multiple users playing different roles within the tasks. Our tests show that perceived utility may be very different for users performing different roles within a task. When determining what network resources are required to complete a task, it may be necessary to base predictions on the most constrained role within a task.

While carrying out a task, a user may adjust to a task and network condition combination. Our tests show that user perception of utility changes as a user repeats tasks over the same network conditions. Users may benefit by starting to talk over a VoIP connection before beginning a task. Users may also benefit by training over simulated bad network conditions.

References

[1] Stehle, E.; Shevertalov, M.; deGrandis, P.; Mancoridis, S.; Kam, M., “Task Dependency of User Perceived Utility in Autonomic VoIP Systems,” *Autonomic and Autonomous Systems, 2008. ICAS 2008. Fourth International Conference on* , pp.248-254, 16-21 March 2008

[2] Karlsson, M.; Covell, M., “Dynamic Black-Box Performance Model Estimation for Self-Tuning Regulators,” *Autonomic Computing, 2005. ICAC 2005. Proceedings. Second International Conference on*, pp.172-182, 13-16 June 2005

[3] Kephart, J.O.; Walsh, W.E., “An artificial intelligence perspective on autonomic computing policies,” *Policies for Distributed Systems and Networks, 2004. POLICY 2004. Proceedings. Fifth IEEE International Workshop on* , pp. 3-12, 7-9 June 2004

[4] Walsh, W.E.; Tesauro, G.; Kephart, J.O.; Das, R., “Utility functions in autonomic systems,” *Autonomic Computing, 2004. Proceedings. International Conference on*, pp. 70-77, 17-18 May 2004

[5] Sterritt, R., “Pulse monitoring: extending the health-check for the autonomic grid,” *Industrial Informatics, 2003. INDIN 2003. Proceedings. IEEE International Conference on* , pp. 433-440, 21-24 Aug. 2003

[6] Sterritt, R; Bustard, D “A health-check model for autonomic systems based on a pulse monitor” *Knowl. Eng. Rev.*, Cambridge University Press , vol.21, no.3pp.195-204, 2006

[7] Hong-Linh Truong; Fahringer, T.; Nerieri, F.; Dustdar, S., “Performance metrics and ontology for describing performance data of grid workflows,” *Cluster Computing and the Grid, 2005. CCGrid 2005. IEEE International Symposium on* , vol.1, pp. 301-308 Vol. 1, 9-12 May 2005

[8] Markopoulou, A.P.; Tobagi, F.A.; Karam, M.J., “Assessment of VoIP quality over Internet backbones,” *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE* , vol.1, pp. 150-159 vol.1, 2002

[9] Hall, T. “Objective Speech Quality Measures for Internet Telephony” *Proceedings of SPIE Voice over IP VoIP Technology*, vol. 4522, pp. 128-136, July 2001

[10] Johannesson, N.O., “The ETSI computation model: a tool for transmission planning of telephone networks,” *Communications Magazine, IEEE* , vol.35, no.1, pp.70-79, Jan 1997

[11] Cole, R.G.; Rosenbluth, J.H., “Voice Over IP Performance Monitoring” *SIGCOMM Computer Communication* Rev. 31, 2, Apr. 2001

[12] Kitawaki, N.; Itoh, K., “Pure delay effects on speech quality in telecommunications,” *Selected Areas in Communications, IEEE Journal on* , vol.9, no.4, pp.586-593, May 1991

[13] Rix, A.W.; Beerends, J.G.; Hollier, M.P.; Hekstra, A.P., “Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs,” *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP ’01). 2001 IEEE International Conference on* , vol.2, pp.749-752 vol.2, 2001

[14] Kitawaki, N., “Perceptual QoS assessment methodologies for coded speech in networks,” *Speech Coding, 2002, IEEE Workshop Proceedings.* , pp. 80-82, 6-9 Oct. 2002

6. Appendix

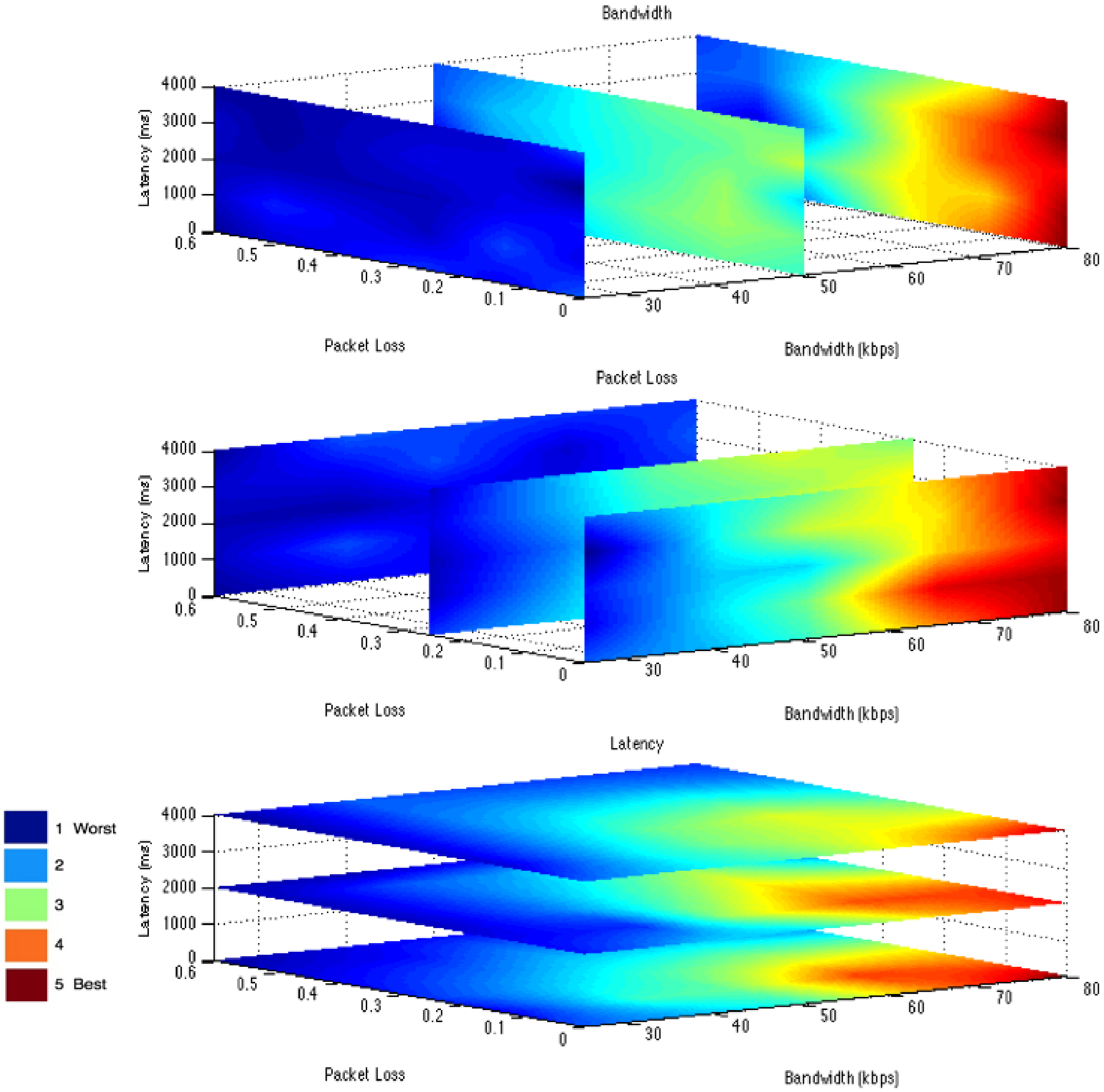


Figure 7. Simple Information Exchange

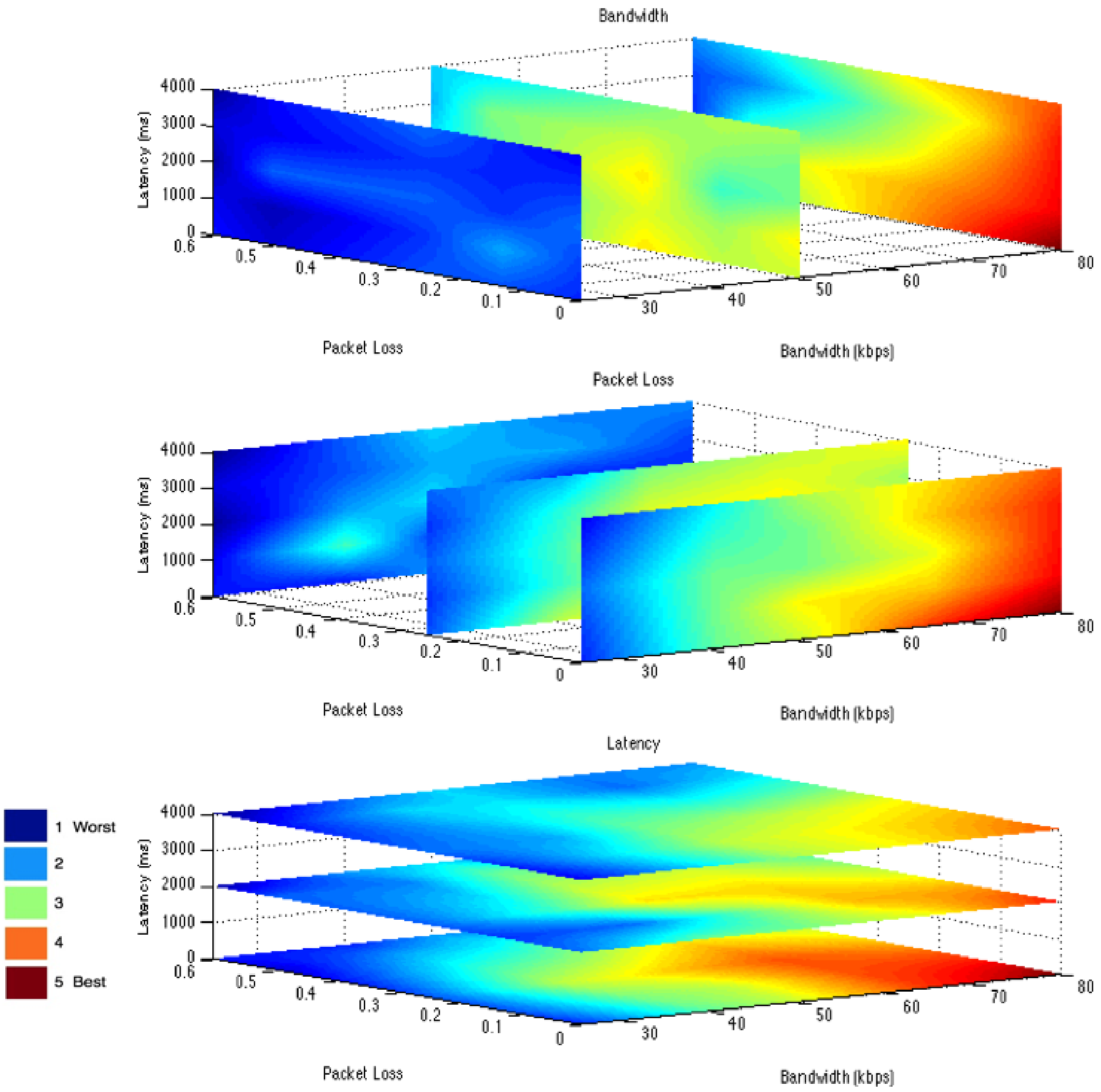


Figure 8. Time Sensitive Collaboration

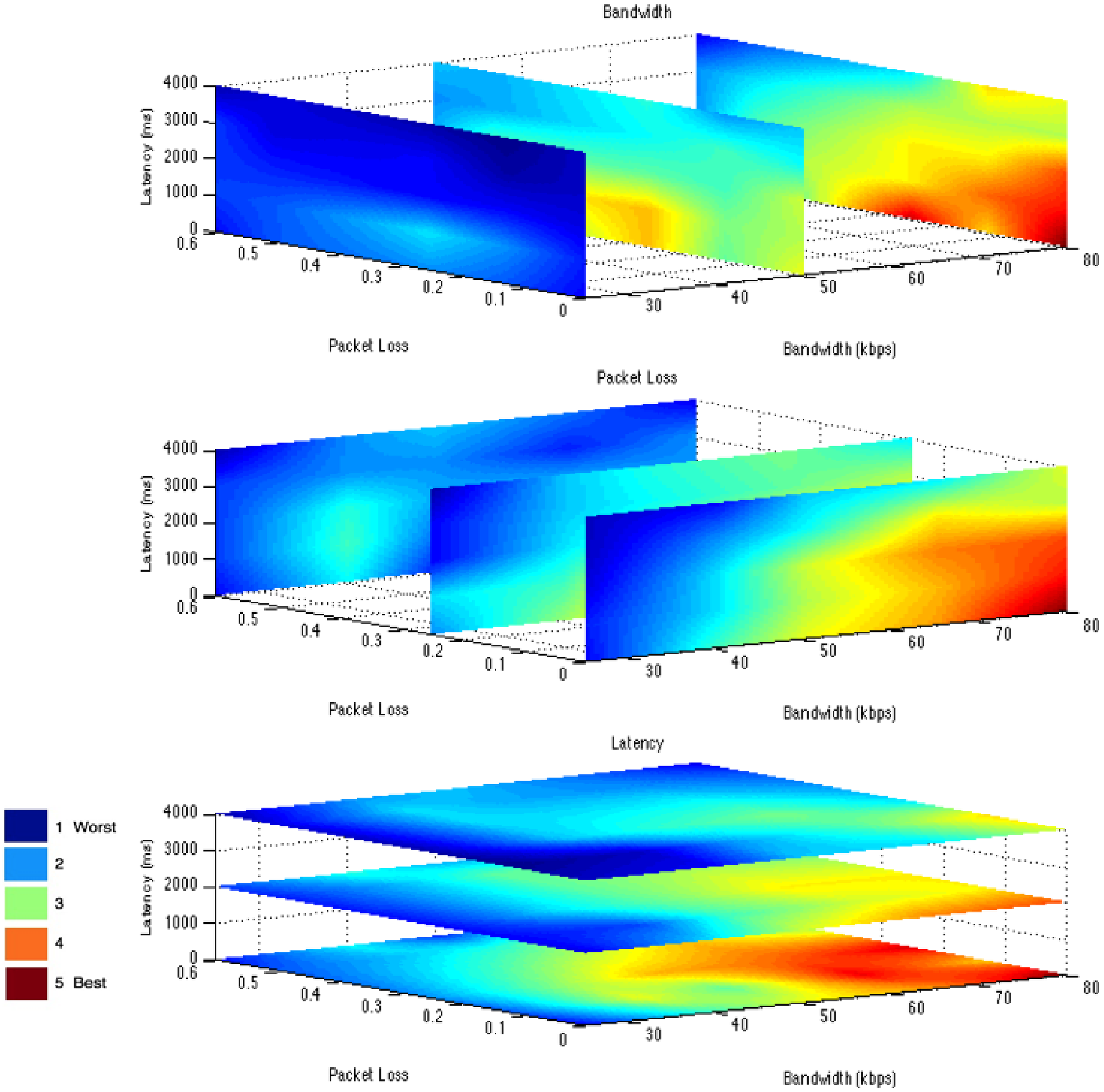
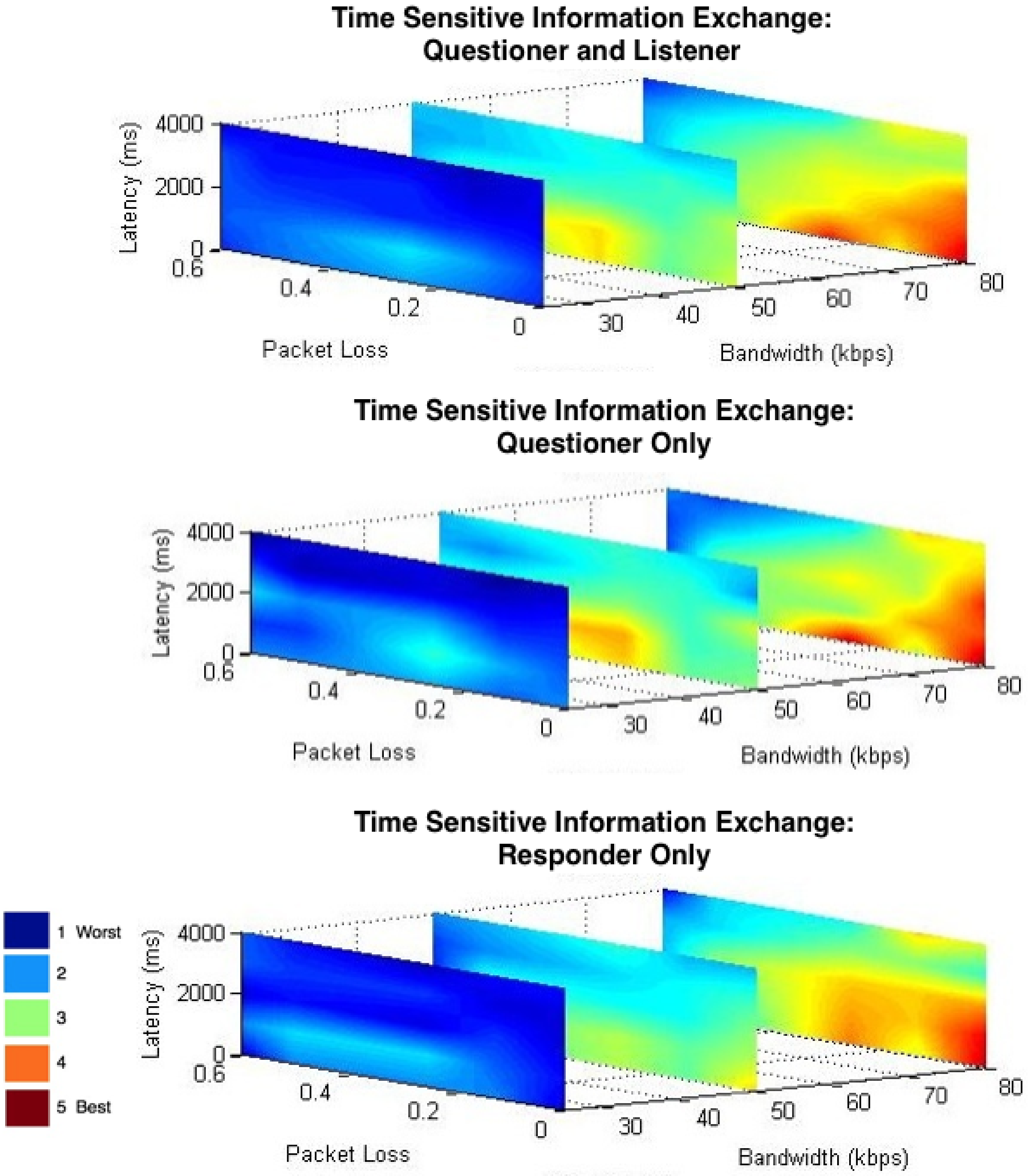


Figure 9. Time Sensitive Information Exchange



Time Sensitive Information Exchange: Questioner and Listener

Time Sensitive Information Exchange: Questioner Only

Time Sensitive Information Exchange: Responder Only

Figure 10. Time Sensitive Information Exchange Split by Questioner and Responder

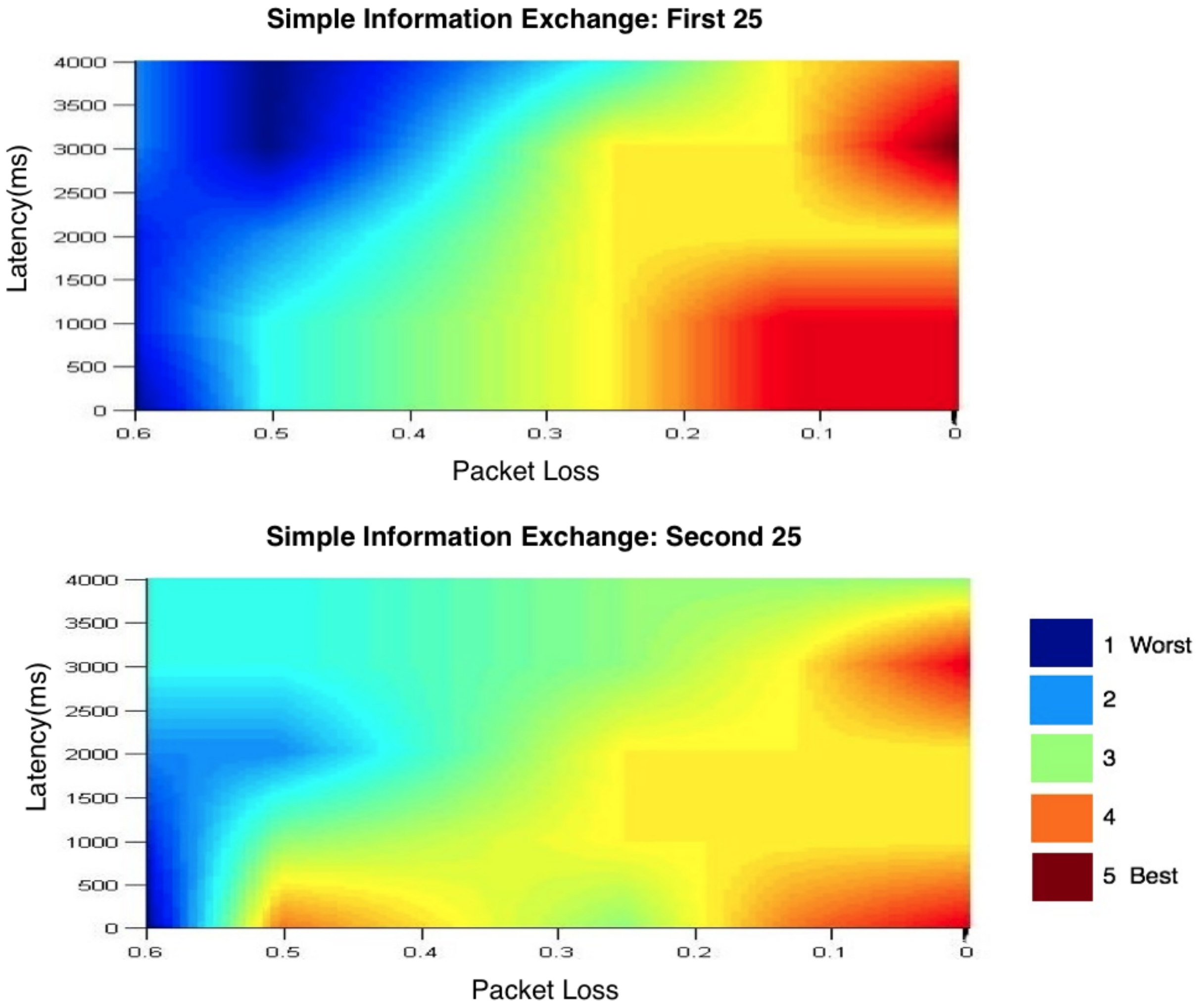


Figure 11. Simple Information Exchange User-Adjustment Tests

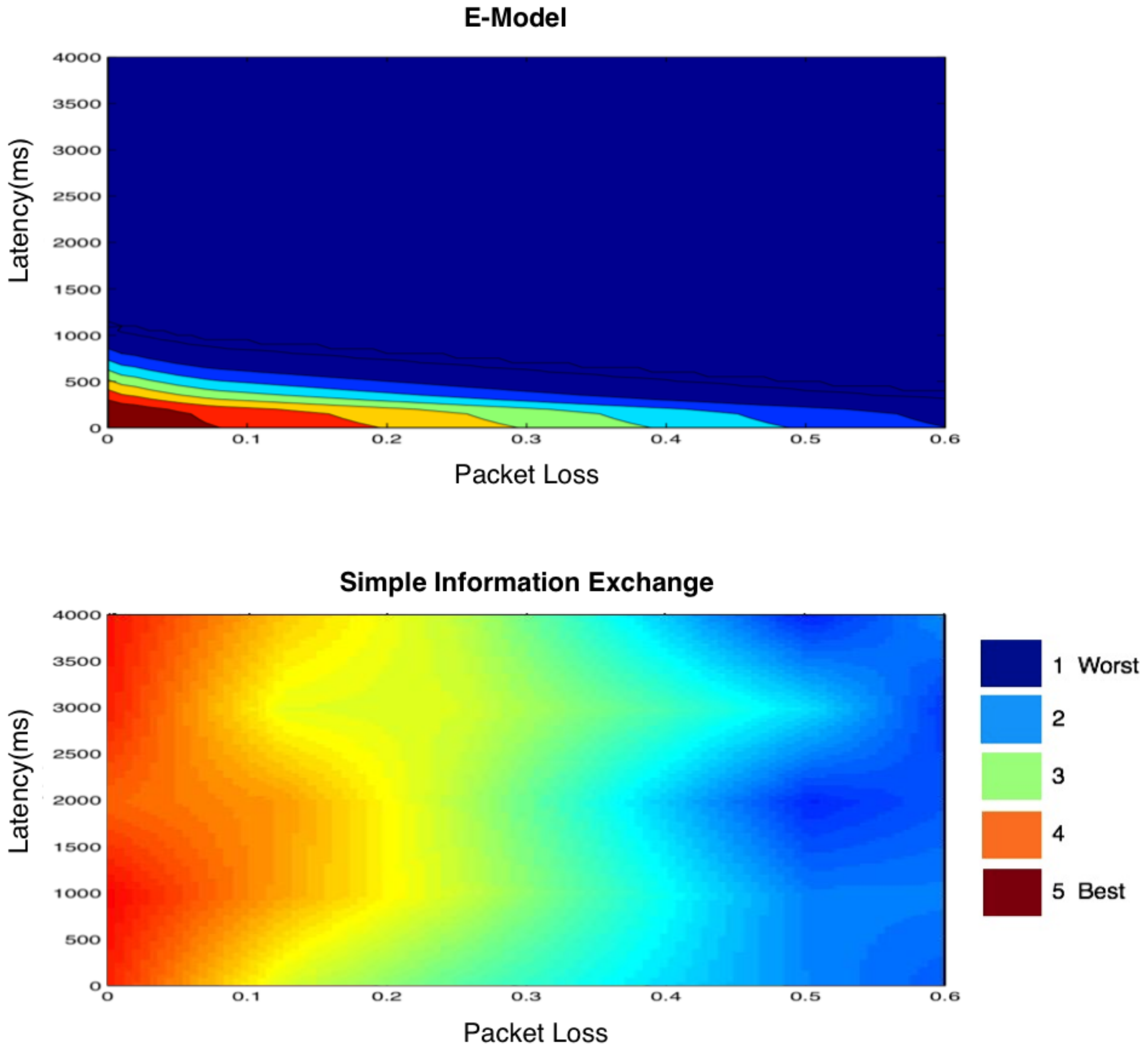


Figure 12. E-Model vs Simple Information Exchange

Analysis of Enhanced Access Selection Methods and End-User Perception in Multi-operator Environment

Petteri Poyhonen
Nokia Siemens Networks
Espoo, Finland
Petteri.Poyhonen@nsn.com

Jan Markendahl
Wireless@KTH
Royal Institute of Technology
Stockholm, Sweden
Jan.Markendahl@radio.kth.se

Ove Strandberg
Nokia Siemens Networks
Espoo, Finland
Ove.Strandberg@nsn.com

Janne Tuononen
Nokia Siemens Networks
Espoo, Finland
Janne.Tuononen@nsn.com

Martin Johnsson
Ericsson AB
Stockholm, Sweden
Martin.Johnsson@ericsson.com

Abstract

Nowadays, the access selection methods are done based on operator incentives using static and predefined rules. But once the number of deployed access technologies and mobile services is increasing, then today's access selection practices are not adequate anymore. We need to think about end-user preferences in a decision making process. Two new distributed decision making algorithms called the Network Centric and the Terminal Centric algorithms exploiting Ambient Networks mechanisms are presented. These algorithms use a richer set of constrains and rules. In order to evaluate these algorithms from an end-user perspective, we introduce a new performance index called the User Satisfaction Index. We present our simulation model, performance results and analysis. The results indicate that the cooperation between networks increases the network utilization and service availability. The benefits from an end-user perspective are expressed using the proposed User Satisfaction Index. Finally, we discuss on further challenges of a decision making for information centric networking.

KEYWORDS: Access selection algorithm, Network cooperation, User experience, User satisfaction index, Network composition

1. Introduction

The recent achievement in technology domain and in business concepts, have resulted large increase in the use of wireless broadband that can be observe today. For instance, deployment of High-Speed Downlink Packet Access (HSDPA) technology in the 3G networks has provided the basic capacity for large scale usage of mobile broadband. As an example from the prolong we mention, Wireless LAN (WLAN) access at hotspots, which have been available for many years at quite high cost, but nowadays they are offered at very low cost, for "free" or included in some service bundle. In general, the introduction of very attractive flat rate pricing for the mobile broadband access is one of the key reasons for the "mobile data explosion".

However, this large increase in traffic will put very high requirements on the availability of low cost high capacity wireless networks implying that additional capacity and investments are needed. While new radio access technologies like Long-Term Evolution (LTE) will contribute substantially to the needed reduction of network costs, mobile network operators will need to develop and consider new deployment and cooperation concepts for network sharing, roaming at a national and local level, reuse private networks and off-loading of traffic to low cost networks. Especially off-loading and exploitation of available accesses in a multi-access environment require more sophisticated access selection means in order to hide the extra complexity introduced by a diverse networking environment for the end-users.

In this paper we will discuss and analyze cooperation between different types of radio access networks and extend the simulation results presented in The Third International Conference on Systems and Networks Communications (ICSNC 2008) [1]. The cooperation between operators both result in lower network costs as well as increased network utilization and availability for the end-users when enhanced decision making is used.

Someone may ask *"what's new here?"*. Operator cooperation and use of multi-radio access technology have existed for many years. Many operators already use network sharing (e.g. based on [2]) and national roaming. Networks with different types of Radio Access Technologys (RATs) have also existed for many years. The principles of Multi-Radio Resource Management (MRRM) have been researched, designed and tested for many years and the performance gains are well known. Also interworking between 3G and WLAN systems [3] is standardized. So, it sounds that good justification is needed to make this topic relevant and *"new"*.

One part of the justification of extended network cooperation is related to service availability and coverage. Operators traditionally allow own customers to access their own network only. In case when the own network does not have coverage the own customer simply cannot connect to other available networks on the coverage. For international roaming the case is different, the user terminal switches to another operator networks as soon as the coverage is reduced for the current operator. In this sense visitors have better service availability due to the international roaming agreements. Hence, the type of cooperation we are proposing is more or less national roaming applied at a local and regional level [4] [5] [6]. In general the principle should be that "anyone" can connect to "any" network [7] [8] [9]. However, in this paper we assume that users have agreements (subscriptions) with some kind of service provider [5]. Typically this is a network operator but it could also be a virtual operator (without own access network resources), a service provider or a trusted third party, e.g. a credit card company.

Second part of the justification of increased cooperation is related to traffic, revenues and network costs. During the last two years there has been a very large increase of usage of mobile data using wide area networks, up to 300-500% . The data service problem for the operators is that although the traffic increases substantially, in contrast to the voices service (and charging per minute of use) where the revenues and traffic increase coherently, increase in the revenues is falling behind [10], due to dominating flat rate charging. The revenue per MB of data is around 1 euro for voice and 0,01 euro for "data" and for some operators the mobile data is 60-80% of the traffic but the corresponding revenues are only 10%.

The operators are challenged by what we can call a "revenue gap". Hence, the operators must focus a lot on cost control and deployment of "low cost" networks in order to meet the increased demand of mobile data services. The different types of cooperation mentioned above can all contribute to this cost reduction although it will not "solve" the whole problem. In addition to the possible cost savings a number of other aspects need to be considered and analyzed in order to both understand and exploit the possible benefits of cooperation:

- the structure and organization of a market with a multitude of network operators, service providers and different types of third parties and middlemen
- the impact on markets and competition due to the facts that end-users more freely can choose between many providers
- the types of business relations and agreements between providers
- the algorithms used for selection of network and radio access technology
- the potential improvements in network utilization and service availability
- the user experience of the increased service availability

The three first aspects have been discussed and analyzed in both public deliverables [4] [11] [12] of the EU project Ambient Networks [13] as well as in papers presented at international conferences [14] [15]. In this paper we will focus on the three last aspects and we also further extend the analysis and results presented in [1] [16] [17].

In this paper we introduce a distributed decision making algorithm that both enables and exploit the cooperation, i.e. the algorithms used for establishing the cooperation and for performing the access selection and handover. We then evaluate the impacts from both as network perspective, in terms of network utilization and service availability, as well from a user perspective. Hence, we can identify three research questions related to:

- Selection of decision making parameters for access selection
- What kind of technical benefits new decision making algorithms potentially result
- How improved network performance and service availability translate into user satisfaction

For the user aspects of the network cooperation we want to analyze the impact of different levels of service availability on the user experience.

The rest of this paper is organized as follows; in Section 2, we outline related work, introduce most relevant Ambient Networks (AN) concepts for distributed decision making and address what is missing. User experience and the User Satisfaction Index (USI) model are explained in Section 3. The decision making algorithm and the used handover model are illustrated in Section 4 and an example of evaluating the discovered radio accesses is given. The simulation model and settings used in our performance evaluation tests are described in Section 5. Section 6 presents the technical simulation experiment results and analyse them from network and operator viewpoint. Section 7 presents the user satisfaction analysis based on the presented technical results. In Section 8, we discuss on further challenges of a decision making in *Future Internet* like networking environments. Finally, in Section 9, we present the conclusions of our work.

2. Related work

2.1. Operator and network cooperation

The integrated European Union (EU) project AN proposes a framework for dynamic cooperation between networks and business entities called *Network Composition* [4] [18] including both business and technical aspects [12]. Network sharing, international and national roaming are all well known examples of a cooperation between operators. One of the innovations with the Ambient Networks is that the cooperation between operators can be established dynamically, e.g. roaming between different local or national networks is possible without pre-negotiated agreements.

The framework for *Network Composition* identifies different levels of cooperation between networks, spanning from more looser/weaker cooperation on towards that two composing networks even gets integrated into just one network. This is to give support for the many various ways the network owners would like to cooperate with each other, and which would depend not only on the specific situation but also for example what kind of networks that would be composed, the legal and business status, etc.

The level of cooperation is modelled out from in what way a certain network has the right to access and control a certain Resource in a network. Before composition, a network has generally full control over its own Resources. After composition, the network might have rights to another network to access and control a certain set of Resources out of its own network, as well as has been given the right to access and control Resources in the other network. If shared control over a certain Resource is agreed, there is a need to create a new virtual control plane in order to manage and operate this shared control.

2.2. Multi-radio resource management and access selection

The AN project also considers Multi-Radio Architecture (MRA) where cooperation between different types of RATs is considered [19]. One example is a design of strategies and algorithms for Radio Resource Management (RRM) for a joint control of heterogeneous radio access networks, e.g. WLAN, Global System for Mobile communications (GSM) and Universal Mobile Telecommunications System (UMTS).

A number of projects have focused on network cooperation and resource control in heterogeneous networks [20] [21] [22]. In the FP5 project Monasidre a service and network resource management platform was developed focusing on radio access networks. The FP6 projects Everest and Aroma have been focused on strategies for efficient radio resources management in heterogeneous networks for support of end-to-end QoS. For both these project "Common RRM" has been a key feature for the management of radio access technologies. The Aroma project also included techno-economic evaluation of micro-cell and WLAN usage within 3G networks.

In the AN project not only multi-radio access is considered, but also multi-operator aspects which are one main theme in this paper. The AN MRRM provided common means to manage and control different radio access resources over network boundaries when the network cooperation were supported [23]. In addition, this paper makes use of the techno-economic modeling developed and used in the AN project [12].

2.3. Analysis of user experience

The authors of this paper have proposed the use of a set of performance metrics as a tool to measure user satisfaction in network cooperation compared to single operator networks. For this modeling and analysis of the user experience we have used the approach proposed by Pohjola & Kilkki [24]. In their proposed methodology value creation of services is modeled together with how users behave and put value on the experience. The assumptions on user perception and rating of service quality are based on the findings by Twersky & Kahneman [25]. They describe the use of Expected Experience and Expected Value function to represent user happiness. If the expected value increases or decreases from the expected value in the user happiness function an increment results in less additional "positive" experience compared to a larger "negative" experience for a corresponding decrement. This leads to different shape of "utility curves", i.e. how the utility for a user depends on different parameters of a service. Examples of utility functions as a function of bit rate are presented by Sachs et al [26].

This kind of "behavioral economics" proposed by Twersky & Kahneman has also been used by Mitomo et al [27] for analysis of consumer preferences for flat rate. Lambrecht & Skiera [28] have investigated consumer behavior related to flat rate charging schemes for Internet services.

Edell & Varayia [29] [30] present trial results on how users value different qualities of service (rate) for fixed Internet (broadband) access. Compared to the work by Edell & Varayia we extend the analysis to present and future wireless broadband services

User satisfaction in terms of throughput for multimedia traffic is analyzed in [31]. Badia et al [32] model the user

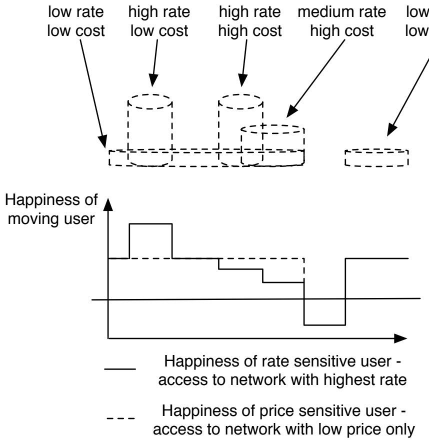

low rate
low cost
high rate
low cost
high rate
high cost
medium rate
high cost
low rate
low cost
Happiness of moving user
Happiness of rate sensitive user - access to network with highest rate
Happiness of price sensitive user - access to network with low price only

Figure 1. Sensitivity and happiness.

satisfaction taking into account requested Quality of Service (QoS), data rate and also price. This model enables analysis of impact of resource allocation on operator revenues.

Customer value related to value creation of services and products is the staring point of the "Value model" proposed by Lindstedt et al [33] [34]. The customer value is expressed as the ratio of *satisfaction of needs* and the *use of resources*. The resources can be time, money, efforts, etc.

Noriaki Kano proposed a customer satisfaction model (Kano Model) that challenged traditional approaches [35] [36]. Different service attributes are not seen as equal by the customers, "some attributes produce higher levels of satisfaction than others". Different consumers will value different attributes differently and hence different consumer categories can be identified.

3. User experience

In this section we will describe our approach for modeling and evaluation of the user experience. The results are based both on simulations, where we estimate the service availability for different levels of cooperation between networks, and on a user survey with focus on the perceived importance of different parameters.

First we will provide an overall description (an illustration) of our approach. Next, based on established modeling and methodology, we will discuss different ways to describe and model the user experience of services and the related customer satisfaction or dissatisfaction.

3.1. User perception and network cooperation

Cooperation between wireless operators is usually seen as means to reduce network costs. Network cooperation may result in a wide spectrum of advantages for both the users and the operators. Network sharing will result in lower costs for the cooperating partners. Such lower costs may or may not result in lower prices for the users. In this paper our main interest is the potential to improve the service coverage and quality.

For the users the cooperation between networks with different coverage will lead to an increased availability and perceived service quality and hence more satisfied users. This may lead to increased usage as well as willingness to pay for the services and hence potentially more revenues. More satisfied users will be more loyal to the operator and hence cost related to churn may be reduced [37].

As an illustration we will assume that the "User happiness" is related to two factors only; the price and the quality of the connectivity service. We consider a set of users and an analysis over a number of time units T_j. For each time unit of duration T_j we consider the perceived service quality Q_j and the price P_j. We assume that the users in every time unit are able to select a new type of network (assuming that the network cooperation allows that), each with its own quality and price. Equation (1) shows how "User happiness" is calculated for one user.

$$UserHappiness = \sum_j Q_j T_j / P_j \tag{1}$$

The "User happiness" increases with increasing service availability and quality and with decreasing price. A simple illustration on the use of this modeling approach is shown in Figure 1. Moving users can be connected to networks with different data rate and price characteristics. Wide cylinders indicate networks with wide area coverage and the heights indicate capacity (or average data rate). High data rate and low price imply a high "User happiness". Disconnection results in a low or even a negative "User happiness".

3.2. General aspects and modeling of user perception

In Section 3.3 we will introduce a more general performance metric called User Satisfaction Index. The USI takes into account both the service quality, availability and the price. In our simulation experiment, to be presented in Section 7, the end-user perception of the service (USI) is calculated based on the technical parameters, e.g. the number of connected mobile nodes.

One way to describe the user value or satisfaction is to consider the total utility for the user, the "price" paid and the corresponding production cost and profit for the service provider. From the consumer perspective there is a "surplus" if the perceived user utility exceeds the price. From the provider perspective the offered price is the production cost plus some profit. The profit is kept secret to the consumers and the consumers want to keep the consumer surplus secret to the producers. If the producer finds out that the consumer surplus is very high, then the price can be increased without any major complaints and hence, the profit is increased. In summary, the user (consumer) tries to maximize the consumer surplus and the producer tries to maximize the profit (i.e. price - cost). In addition, With the utility model described above the consumer surplus can be described as "added value" in different dimensions; e.g. access to a service or product, time saving, ease of use, convenience, etc. However, the "price" in this utility model usually is related to money only, but the "cost" for the consumer could be extended to include other aspects where the consumer need to "pay" in other ways, e.g. by allocating time, to provide own work or different sorts of "inconvenience".

This type of reasoning is one of the main characteristics of the "Value model" proposed by Lindstedt et al [33]. The $Customer\ Value$ is expressed as a ratio of "Satisfaction of needs" (SN) and the "Use of resources" (UR); Equation (2). The resources can be any combination or function of time (t), money (m) or effort (e), i.e. $f(t, m, e)$.

$$Customer\ Value = SN/UR \qquad (2)$$

The function ($f(t, m, e)$) can most often not easily be described in general terms. Assume for example that the function is a product or a sum of the individual functions $f(t)$, $f(m)$ and $f(e)$. Assuming the product of separate functions, then a very small (or zero) value, e.g. price = 0, would lead to a very large (or infinite) customer value. Assuming a sum of individual functions combination would requires some form of "weighting", and this would probably be very case dependent.

Noriaki Kano has developed an approach based on "Attractive Quality Creation" that usually is referred to as the "Kano Model". This model has been used for development of new products and services and to determine market strategies.

When this approach was presented Kano challenged the traditional Customer Satisfaction Models based on an assumption that "More is better". This assumes that the better the provider can perform on each service attribute the more satisfied the customers will be. This would e.g. imply a more or less linear relationship with different attributes, e.g. if the bit rate of a communication service is increased 10 times then the satisfaction of the customer will increase 10 times.

In the proposed customer satisfaction model (Kano Model) it is assumed that the performance on product and service attributes is not equal in the eyes of the customers. Performance on certain categories attributes produces higher levels of satisfaction than others. In addition, different consumers can value different attributes differently

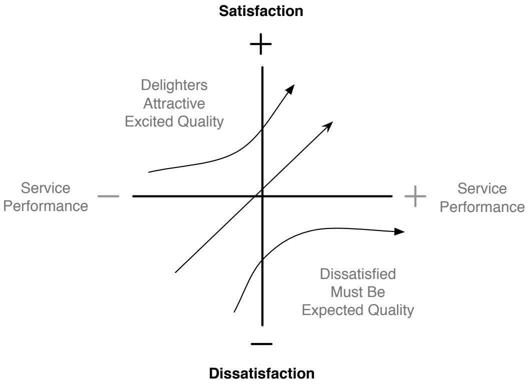


Figure 2. Kano Model.

An illustration is shown in Figure 2 where three types of different customer response to a service is shown [38]. The traditional assumption on customer behavior, i.e. "more is better" is denoted "Satisfier". Another type called "Delighter" is a customer that appreciates more attributes (and performance). Finally, a customer that requires that the service always includes all possible attributes and have the best possible performance, i.e. there are a lot of "must bes" , is called a "Dissatisfier".

In the Ofcom studies of consumer experience and decision making [39] [40] a number of factors was identified describing different aspects of user satisfaction. Some factors are called "emotional", e.g. the trustworthiness of the brand and how highly other people rate the brand of the operator. Most listed factors can be denoted "rational/tangible" and can be grouped into different categories:

- Network related factors: reliability of coverage, ease of use of network services, reliability and speed of the connection.
- Factors related to the price and service offers: low cost, amount of data that can downloaded, ability to get bundled offers and "value for money"
- Factors related to customer care: technical support and customer service

3.3. Our approach - USI model

In our modeling approach we have used two starting points i) users have some level of expectation about the service availability and quality and ii) the impact of "no service", i.e. disconnection needs to be taken into account. For the aspect of "expected experience" we have used the Twersky & Kahneman findings about changes in service levels. If the quality is decreased by some amount this has a much bigger negative impact than the corresponding positive im-

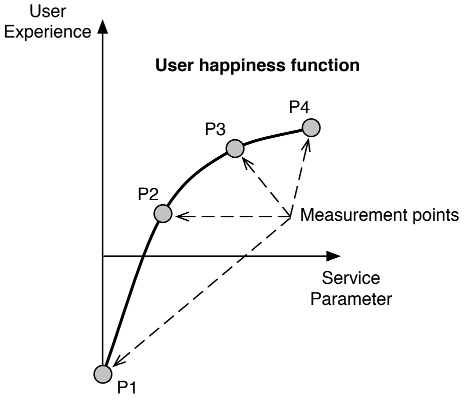


Figure 3. User happiness function.

pact of an equally large positive change. When it comes to disconnection we model this with a negative value of the experience. A zero value for the user experience would imply that the user "does not care".

In our modeling we define a performance index called User Satisfaction Index. The USI model is based on the user happiness function of 4 levels. These levels represent different perceived service qualities using different P values; $P1$, $P2$, $P3$ and $P4$, see Figure 3.

The value $P3$ represents the user happiness when the service parameter has the expected value. $P2$ and $P4$ represent the user experience when the service parameter is lower or higher than the expected value. $P1$ corresponds to the case when there is no service, i.e. no connection. As Figure 3 illustrates, the better the expectations are satisfied, the happier the user will be.

Four levels were selected based on the assumption that a user always has a certain expectation level based on which the perceived service quality can be evaluated. This expectation level can vary based on various things like a service type, user's earlier experiences and so on. This assumption alone leads to 3 different levels and is sufficient as such to model the quality sensitivity. However, in order to model also the connectivity sensitivity, we need to differentiate disconnections, when there is no perceived service, from the situations where a user perceives the service quality that is worse than expected. So in order to model both the connectivity and the quality sensitivity, 4 different levels are required. The model could be extended by introducing more measurement points to model different perceived service qualities, but for the sake of simplicity, we settled down to 4 levels, which was a trade off between accuracy and simplicity. It should be noted that some (non-elastic) services might not tolerate large range of fluctuation in the service quality level resulting in an unusable service if only the perceived service quality corresponds $P2$.

Equation (3) presents the USI for user i, where K is a number of services, X ,Y, Z and W are numbers of P value measurements. α, β, χ and δ are P value weights and $Cost$ is the end user price per data unit. Equation (4) is used to calculate the overall USI of all users.

Focus	α	β	χ	δ
Connection	-1	1	1	1
Connection/Quality	-1	0.25	1	1.4

Table 1. P value weight sets.

$$USI_i = \sum_{j=1}^{K}(\alpha \sum_{i=1}^{X} P1_{ij} + \beta \sum_{i=1}^{Y} P2_{ij}/Cost_j + \chi \sum_{i=1}^{Z} P3_{ij}/Cost_j + \delta \sum_{i=1}^{W} P4_{ij}/Cost_j) \quad (3)$$

$$USI_{all} = \sum_{i=1}^{L} USI_i \quad (4)$$

In the USI analysis, two different P value weight sets as represented in Table 1 are considered. The first set has its focus on connectivity and it does not differentiate between $P2$, $P3$ and $P4$ values and therefore the same weight value is used for the corresponding weights β, χ and δ. The second set extends the first one by differentiating the quality levels and as a result of this, different weight values are assigned for $P2$, $P3$ and $P4$ values.

The first P value weight set corresponds the services tolerating quite well short temporal connection breaks like a web surfing. Respectively, the second P value weight set is typical for the real time services.

3.4. User survey for the USI - data collection

In order to verify the modeling assumptions used for the estimation of the USI metric a user survey was conducted on user perception of services. The survey included three parts

- Part one consists of one open question "what are the most important aspects for usage of wireless broadband and selection of service offers?".
- The second part includes rating of different statements on how the person would perceive different levels of service quality, e.g. availability and delivered data rates.
- In the third part the persons were asked to rate "the attractiveness" of different service offers for wireless

broadband access with different prices, data rates and amounts of usage.

30 persons participated in this "small" survey. Two types of users were included in the survey; telecom people (students at technical university) and "ordinary users" (non-engineers). The participants were asked about their experience of wireless broadband access and where the access was used: at home, at the office/school or in public places. Most participants used some WLAN and/or 3G wireless access on a regular basis.

The objective of part one was to confirm that our "assumed" parameters were the ones that were considered important when the service quality or service offers were evaluated. People were asked to list parameters that were considered important and to provide a motivation. We counted how many persons did mention a specific aspect as having a high degree of importance.

The objective of part two was to get an indication on how people perceive service quality and how they "value" different aspects. People were asked to rate (from -10 to +10) how they did perceive service availability and the delivered data rate assuming a specific value of the expected data rate.

The objective of part three was to get some insight about reasoning when choosing between different offers, how trade-offs are made and what parameters that were considered most important.

4. Algorithm and handover model

In this section, interdependencies between cooperation and decision making are explained to elaborate the importance of network cooperation for a distributed decision making algorithm and to show how diverse environment the decision making should cope with. The decision making algorithm is described with a high level pseudo code and the formula for calculating the cell rank values based on the different constraints is presented. The handover model used in the simulations and its state machine is explained including different kinds of delay contributing to the overall handover execution time as perceived by an end-user.

4.1. Cooperation and decision making

Cooperation can be characterized with 2 different types of agreements: Horizontal and vertical agreements. Horizontal agreements represent a cooperation between network providers. For example, when service continuity is preferred for an existing connection, it requires a horizontal agreement between old and new network providers. Naturally, overlapping operator coverage areas provide a good base for the cooperation between operators to support (seamless) inter-operator HandOvers (HOs) and load balancing.

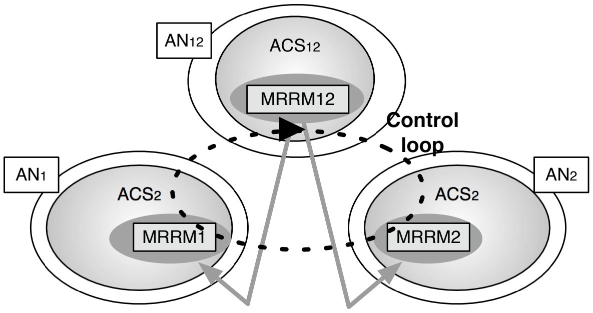


Figure 4. Control Sharing of the MRRM control functionalities.

Vertical agreements are used between network and service providers. This type of agreement represent a cooperation based on which for instance information is collected from a service provider to be taken into account in a distributed decision making process.

Let us consider an example of *Network Composition* when being applied to MRRM, the trade-offs in regard of the performance of access selection, and which should be considered when composing two ANs.

In Figure 4 we let the two networks $AN1$ and $AN2$ compose in order to share the control of their MRRMs, resulting in that a new virtual $ACS12$ is created with a virtualized $MRRM12$ executing within this Ambient Control Space (ACS). For its implementation, it is likely that each of the two composing ANs will instantiate a virtualized MRRM, which will communicate with each other to provide the shared control of the underlying MRRMs. Thus, through $MRRM12$, the control of access selection is distributed and shared, with the capability to select any of the available access networks (subject to their respective status as described above). It should then be observed that due to the distributed aspect of decision making, the time of the control loop for access selection will be extended, and will not be able to respond as quick as if control was made locally (but then of course without the possibility to roam seamlessly between the two ANs). The implementation of a distributed algorithm depends on the used composition type. For instance, in case of delegated MRRM control, a decision making and access selection is more like "centralized" solution where we should expect the time for the control loop that should be close to the time for the local control loops in the non-composed ANs.

A distributed decision making framework should be able to operate on top of diverse business landscape where technical agreements between networks and players like Ser-

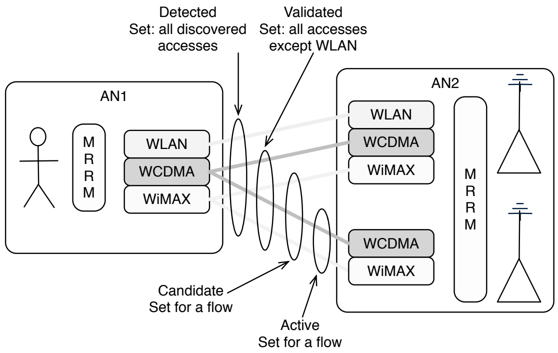


Figure 5. MRRM access sets.

vice Level Agreements (SLAs) realize the business relationships. Algorithm distribution means that relevant information is gathered and used according to existing horizontal/vertical agreements in the decision making.

4.2. Decision making algorithm

Figure 5 shows the MRRM sets based on which the access selection is done in the AN architecture. There are four sets:

- *Detected Set* contains all detected access resources by a terminal
- *Validated Set* contains all access resources from the *Detected Set* that are validated by policy functions and are usable
- *Candidate Set* contains all access resources from the *Validated Set* satisfying the given requirements like the resource requirements of a flow
- *Active Set* contains the selected access resources for a flow

The algorithm is using these sets with a few exceptions. First, we do not use *Validated Set* since our simulation model does not include any special access policies. Secondly, our algorithm uses extended access sets, i.e., there are two different *Candidate Sets* representing both terminal and network preferences. The algorithm execution starts in a terminal based on a trigger generated either in the terminal or network. A terminal is naturally the only entity able to detect what cells are in its coverage. After this, the detected cells are communicated to the network side and where each cell is ranked based on the network's preferences. A terminal does the same for each cell. Finally, both the terminal and network cell ranks are considered together to decide what cell is the best one.

Strategy Name	α	β
Terminal	3	1
Network	1	3
Legacy	1	1

Table 2. Algorithm weights.

Algorithm 1 shows a high level pseudo code of the access evaluation and selection algorithm, which is executed once in a time unit. The algorithm gets a set of mobile nodes as input parameter. After this, the order in which the mobile nodes are processed is randomized. For each mobile node, first the $DetectedSet$ is constructed. This set contains all radio cells a mobile node can detect and which have enough available resources according to the mobile node's demands. Once the $DetectedSet$ is done, then each cell in it is evaluated according to Equation (5) and the resulting cell rank value is added to the $CellRanks$ vector. If the cell with the highest rank is not the one, which the mobile node is currently using, then a handover is performed and the $ActiveSet$ is updated with the new cell info.

```
Input: Set of mobile nodes
Output: Error status
randomize the order of mobile nodes;
foreach Mobile node $i$ do
    Read current mobile node status;
    Update mobile nodes location info;
    Construct $DetectedSet$;
    foreach Cell $j$ in the $DetectedSet$ do
        Calculate cell rank value;
        Add the rank value to the $CellRanks$;
    end
    $BestCell$ = CellWithMaxRank($CellRanks$);
    if CurrentCell != $BestCell$ then
        Perform handover;
        Update the $ActiveSet$;
    end
end
```

Algorithm 1: Update MN states.

In Equation (5), CR_i is the cell rank value for cell i, α is the *Terminal Centric* algorithm weight and respectively β is the weight for the *Network Centric* algorithm. The algorithm assumes that there is N numbers of constraints for terminal (tc) and M numbers of constraints for network (nc).

$$CR_i = \alpha \sum_{j=1}^{N} \lambda_j tc_j + \beta \sum_{j=1}^{M} \kappa_j nc_j \qquad (5)$$

The algorithm weights α and β are adjusted based on the used algorithm strategy. For the *Network Centric* algorithm $\alpha > \beta$ and correspondingly for the *Terminal Centric* one

$\beta > \alpha$. For the legacy, the same weight value was used for both algorithm weights. Table 2 shows the used weights for each strategy.

4.3. Constraints

In the simulation model, all constraints are classified according to two distinct factors; i) based on the value type of a constraint (binary constraint vs. non-binary constraint) and ii) based on how constraint's conditions should be satisfied (hard constraint vs. soft constraint). Table 3 lists the used constraints, their types based on this classification and their constraint weights. Three first constraints are handled in the terminal and the rest are handled in the network side.

As illustrated in Table 3, the constraint specific weights are only defined for the soft constraints. The hard constraints that are used for qualifying evaluated cells according to the constraint's conditions, the weight value 1 is used in the cell rank calculation. The sum of all terminal/network soft constraints is equal to 1.

Signal strength constraint prefers a stronger radio signal. This is perhaps one of the most significant constraints used in legacy systems to perform access selection.

Selection of RAT constraint prefers the discovered cells that are in the current RAT and it is used to minimize inter-RAT HOs. Respectively *Selection of operator* constraint prefers the discovered cells from the current operator.

Cell load levels and *Service load levels* constraints are used for load balancing. The former is used to prioritize the cells with lower load over the highly loaded ones assuming that cells' load levels exceed the load balancing threshold. The latter does the same for service types.

Roaming agreement and *Supported service type* are both binary hard constraints and they are used to disqualify the cells that belongs to the operator either not having a valid Roaming Agreement (RA) or not supporting the requested service type.

4.4. Calculating cell ranks

Let's consider a simple example without using the real values to illustrates how the $CellRank$ vectors are constructed and used. Let us assume that there are 2 cells (a, b) in the $DetectedSet$ and that there are 2 terminal constraints (A, B) and network constraints (C, D). First, both terminal $(tc_A = \lfloor tc_{a,A}, tc_{b,A} \rfloor, tc_B = \lfloor tc_{a,B}, tc_{b,B} \rfloor)$ and network $(nc_C = \lfloor nc_{a,C}, nc_{b,C} \rfloor, nc_D = \lfloor nc_{a,D}, nc_{b,D} \rfloor)$ constraint vectors are constructed. After this, the constraint vectors are normalized, i.e. a vector element value is between 0 and 1, and multiplied by the constraint specific weights and summed together resulting $CandidateSets$ for terminal (Equation (6)) and network (Equation (7)).

Constraint name	Constraint type	Weight
Signal strength	non-binary/soft	0.6
Selection of RAT	binary/soft	0.3
Selection of operator	binary/soft	0.1
Cell load levels	non-binary/soft	0.6
Roaming agreement	binary/hard	1
Supported service type	binary/hard	1
Service load levels	non-binary/soft	0.4

Table 3. Constraint types.

$$T\_CS = [\sum_{i=1}^{2} \lambda_i tc_{i,a}, \sum_{i=1}^{2} \lambda_i tc_{i,b}] \tag{6}$$

$$N\_CS = [\sum_{i=1}^{2} \kappa_i nc_{i,a}, \sum_{i=1}^{2} \kappa_i nc_{i,b}] \tag{7}$$

Next, both $CandidateSets$ are multiplied by the terminal algorithm (α) and the network algorithm (β) weights and summed together resulting the $CellRanks$ vector consisting of two elements, one for cell a (Equation (8)) and another one for cell b (Equation (9)).

$$CR_a = \sum_{j=1}^{2} \alpha \lambda_j tc_{j,a} + \sum_{j=1}^{2} \beta \kappa_j nc_{j,a} \tag{8}$$

$$CR_b = \sum_{j=1}^{2} \alpha \lambda_j tc_{j,b} + \sum_{j=1}^{2} \beta \kappa_j nc_{j,b} \tag{9}$$

The $ActiveSet$ is then constructed based on these cell rank values as illustrated in Equation (10), i.e., the cell with a higher rank value is forming the $ActiveSet$.

$$AS = \max(\sum_{j} CR_j) = CR_k, k \in [a, b] \tag{10}$$

4.5. Handover model

The HO model combines the radio and application connectivity states. The model consists of five states; *disconnected*, *connected*, *session association*, *radio bootstrapping* and *handover execution*. All applications are using the same session association delay of one time unit. For UMTS, it was assumed that this radio technology is attached all the time due to its low power consumption compared to

Transition	Conditions
A	Simulation start-up.
B	Out of coverage. No available access resources.
C	New radio access discovered.
D	New radio access ready.
E	HO finished successfully and a session needs to be (re)associated due to the HO type or due to the application type change during the HO.
F	The discovered new radio bootstrapped successfully.
G	HO finished successfully and no need to re-associate a session.
H	Session (re)associated successfully.
I	HO initiated with a radio bootstrapping.
J	HO failed and the old cell not available.

Table 4. State transition conditions.

WLAN, which is kept de-attached, if not in use. The bootstrapping time of WLAN was set to one time unit.

For HO execution delays, the UMTS and WLAN performance results from [11] were used as base and a “basic” HO type within a single RAT was chosen to last one time unit. Other types (inter-RAT and inter-operator HO) were chosen to last twice as long as the delay of intra-RAT HOs, i.e. two time units.

In the beginning of a simulation, the bootstrapping delays are not used, thus all Mobile Nodes (MNs) are able to move directly into the *connected* state assuming that enough radio access resources and the requested service type were available. So from an end-user perspective, the overall effective HO execution time is the sum of a HO delay, a radio bootstrapping time and a session association time. Table 4 explains under which conditions the state transitions occur in the handover model represented in Figure 6.

Inter-RAT HOs are not supported by the legacy algorithm, which is always forced to perform a radio re-association when switching between RATs resulting a short period of disconnection. The *Network Centric* and *Terminal Centric* algorithms are supporting Inter-RAT HOs inside an operator network and between operators’ networks with the cooperation.

5. Simulation model

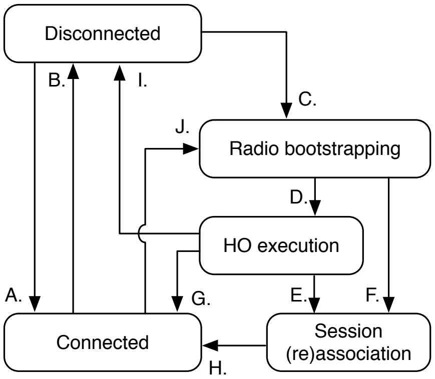


Figure 6. State machine.

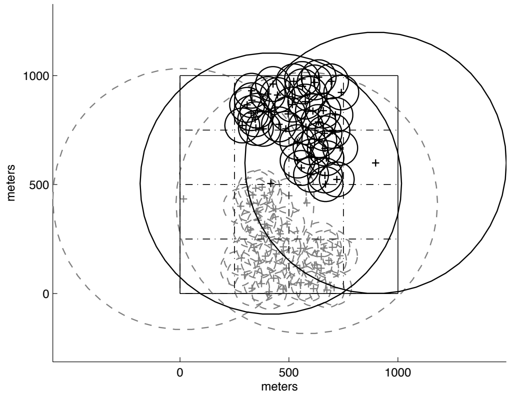


Figure 7. Radio cell topolocy.

In the simulation setup, two operators and two service providers were used. Both operators had a SLA with their service provider and they provided the same RATs, one access network with 45 WLAN cells and another with 2 UMTS cells. WLAN cells had the radius of 80m and UMTS cells had the radius of 600m. The simulation area was one square kilometer. Radio cell deployment in the simulation area is presented in Figure 7 where one operator cells are presented with grey dashed lines and another ones’ with black solid lines. The number of mobile nodes for the scalability tests varied between 100 and 800 MNs, and for other kinds of technical and the USI measurements 300 MNs were used.

For the scalability and the USI tests, the *Network Centric* and legacy algorithms were compared. The *Terminal Centric* algorithm was omitted from these tests, since as clearly showed by other technical evaluations, it finished second after the *Network Centric* algorithm.

The MNs did not follow any particular movement pattern, thus they were moving according to random movement model based on the following limitations;

- Starting locations are randomized based on the uniform distribution,
- The maximum speed of a MN is 10 m/s,

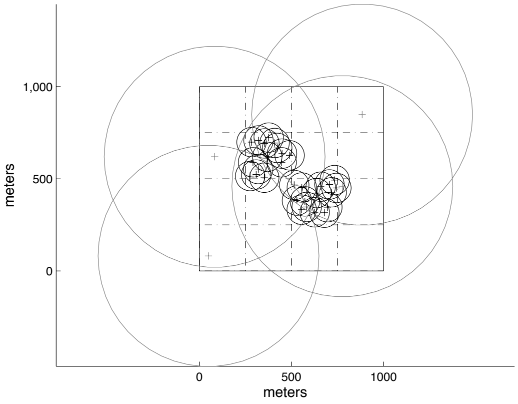


Figure 8. Operator-1's cell topology.

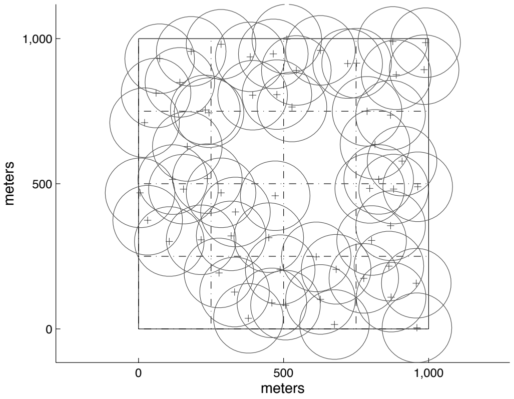


Figure 9. Operator-2's cell topology.

- There are no idle moments for any MN, and
- A random ±90° movement direction change probability is used.

The MNs had an application running all the time and there were two supported service types. Each MN had an application usage vector defining what application type is used and when. These vectors were randomly generated for each MN based on the uniform distribution. If a MN requested the service type that was not available at the MN's current location, then the MN went to the $disconnected$ state and did not reserve any access resources.

Other working assumptions were as follows;

- Each MN supports WLAN and UMTS accesses,
- Each MN supports a HO between and within a RAT and operator when the network side implements such HO,
- Cell loads are measured in terms of an abstract measure for traffic load called *traffic units*,
- Each cell has circle shape coverage area and signal strength S is defined as Equation (11), where d is the distance between a MN and the cell origin and R is the cell radius, and
- Cooperation between operators also includes RA.

$$S = \max[0, 1 - (d/R)] \tag{11}$$

5.1. Competitive Multi-operator Simulation

In parallel with the USI simulations, an additional set of access selection algorithm simulations was run in the business environment consisting two competing operators having overlapping radio access network coverage. The business environment consisted of the operator-1, who was multi-access provider with full coverage of wide-area UMTS over the simulation area and a few dense WLAN hotspots (see Figure 8) and the operator-2, who was a legacy operator providing almost full coverage of short/mid range radio coverage (see Figure 9). The operator-1 multi-access network supported inter-RAT HOs according to the handover model described in Section 4.5. Because of the competition between the operators, in these simulations, HOs between the operator networks weren't possible. However, both operators had roaming agreements to provide access for any mobile node. In other words, all mobile nodes in the simulation could be considered as roamers and mobile nodes could change operator networks via bootstrap (which is more costly than handover).

In the essence, the simulation setup for the competitive multi-operator simulations was pretty much identical to the one presented for the USI simulations in Section 5. However there were some obvious differences, such as the in the business environment and in the radio coverage due to different configuration.

6. Evaluation - network and operator aspects

In this section, we illustrate the technical simulation results in Section 6.1 to show how a distributed decision making algorithm performs against the legacy algorithm and what kind of technical benefits the network cooperation results in. Additional simulation results are presented in Section 6.2 to show how the evaluated algorithms perform in a different kind of networking environment.

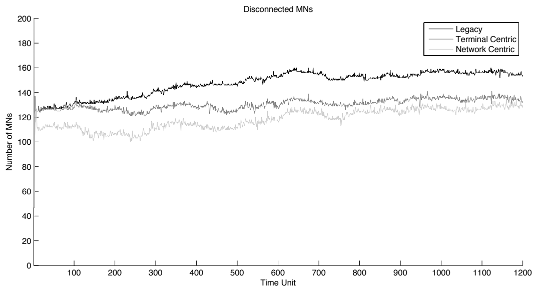


Figure 10. Disconnected MNs - No cooperation.

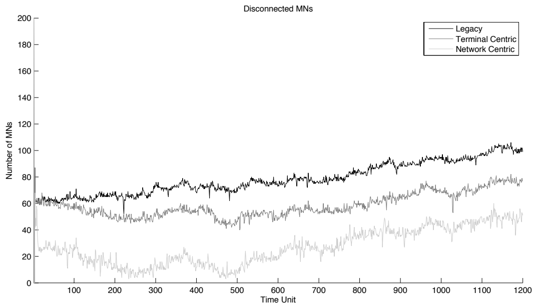


Figure 11. Disconnected MNs - Cooperation.

6.1. Network cooperation analysis

One of the main goals of these simulations is to study what kind of (technical) benefits the new distributed access selection algorithm defined in Section 4 could potentially result with and without the network cooperation. Figure 10 illustrates the disconnectivity measurements for each evaluated algorithm without the cooperation. As can be clearly seen, both the *Network Centric* and the *Terminal Centric* algorithm perform better than the legacy one. An interesting finding is how the *Terminal Centric* performs; in the beginning it is close to the legacy but then it starts to gain gap to the legacy and to approach the performance of the *Network Centric* algorithm. A high number of disconnected MNs is the result of two factors. Firstly, without the network cooperation, a MN is limited to the use of one operator, which allows it to use only a single type of service. Secondly, there occurs temporary congestion in the simulation area, because the available network resources are not uniformly distributed as explained in Section 5, i.e., WLAN hotspots populate only approximately 38% of the simulation area.

The corresponding measurements with the cooperation between operators are showed in Figure 11. All three algorithms perform better than without the cooperation, which was expected. The *Network Centric* and the *Terminal Centric* algorithms can better exploit the network cooperation as showed in the figure. And this is the reason why for instance a gap between the *Network Centric* and the legacy is bigger than it was without the network cooperation.

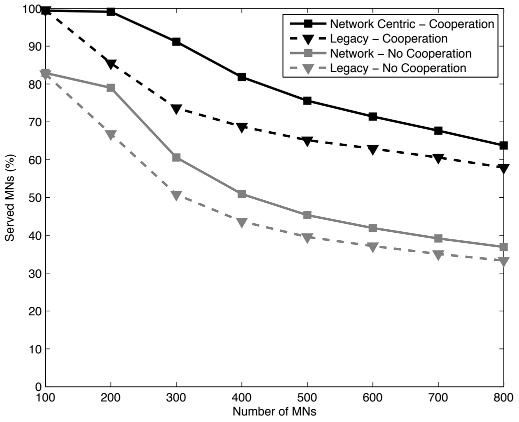


Figure 12. Served MNs.

The scalability measurements for the *Network Centric* and the legacy algorithms are shown in Figure 12. For low network load like with 100 MNs, it does not matter whether the cooperation is used or not. When the network load is increasing, the difference between the cases is becoming clearer. It can be noted that the difference between algorithms is bigger with cooperation than without it. The *Network Centric* algorithm is able to better maintain its capability to serve users with a heavy network load.

After the network load increases over 300 MNs, the differences between the cases with and without the cooperation are getting smaller, but the *Network Centric* algorithm still yields approximately 30% improvement in the network utilization compared to the legacy. Network utilization increases slightly less when the cooperation is not present due to the lack of extended access coverage and supported services. These results indicate that the cooperation results in better effective network capacity and more served users. For an average user, the cooperation means more stable connection and less connection breaks because handovers are supported by the horizontal agreement (cooperation) between the operators.

But all these technical benefits do not become without a *price*; i.e., an increased number of HOs as shown in Table 5. The *Network Centric* algorithm results in the highest number of any kinds of HOs, i.e. over two times more intra-RAT HOs than the legacy case. In practise, this results in more effective load balancing, which can be seen as increased utilization of the available network resources. The *Terminal Centric* algorithm is once again finishing second.

		Intra-RAT	Inter-RAT	Inter-oper.
No Coop				
	Legacy	1.63	0	0
	Terminal	2.77	0.29	0
	Network	3.76	0.36	0
Coop				
	Legacy	4.45	0	0
	Terminal	6.60	0.51	0.41
	Network	9.90	0.56	0.77

Table 5. HO statistics (avg. HOs per MN).

6.2. Competitive multi-operator analysis

Main goal of the competitive multi-operator evaluation was to study how the *Network Centric* and the *Terminal Centric* access selection algorithms perform against the legacy algorithm in the case of multi-access operator and single-radio (legacy) operator.

Key interest and main evaluation metric was set to the network resource utilization in the operator networks, when the requested network load from the MNs was close to and exceeding the combined operator network capacity. In the simulation case, it was also assumed that the service provider has unlimited resources, thus the access network resources were the only limiting factor. Additional results are available in [41] [42].

One additional goal and reason for these simulations was to verify the earlier results [43] [44] that did indicate clear performance gain for the the *Network Centric* and *Terminal Centric* algorithms. While the differences between the algorithms decreased from the earlier results, due to the more detailed modelling of radio access handover and bootstrapping, the trends remained the same.

Figure 13 shows the network resource utilization for the measured three algorithms with 400 MNs and the requested network load of 600 Traffic Units (TUs). The figure also includes fourth graph, which represents the theoretical maximum of the network resource utilization calculated by a linear programming Linear Programming (LP) technique, called Mixed Integer Programming (MIP) [45]. The MIP finds one of the optimal solutions, if it exists, and reports unfeasibility otherwise; therefore it is commonly used in network planning to obtain (theoretical) upper bounds on performance.

The graphs in Figure 13 illustrate the key finding of the simulation results; In the multi-access network, the *Network Centric* and *Terminal Centric* algorithm graphs are clearly closer to the theoretical maximum graph than the graph of legacy algorithm. This indicates that network resource utilization is better for the new algorithms. In addition, the simulation result confirmed expected trend for the single-radio network; studied sophisticated access selection algorithms cannot improve network resource utilization in such environment.

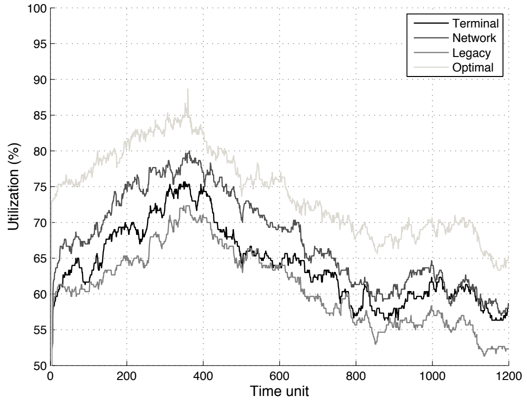


Figure 13. Operator-1's network utilization.

Moreover, Figure 13 shows that in the multi-access network the highest network utilization is gained, when the *Network Centric* algorithm is used. The benefit is quite steady through out the simulation period and can be as high as 10% compared to the legacy algorithm. Further on, the figure shows how the *Terminal Centric* algorithm graph behaves less steadily, but still outperforming the legacy algorithm most of the time.

7. Evaluation - user and usage aspects

The evaluation of user experience is based on two main types of results. First, the results from the user survey on "important aspects" and on the impact of different parameters are presented. Next, the technical results on network utilization and service availability presented in presented in Section 6.1 are interpreted using the User Satisfaction Index model introduced in Section 3.

7.1. Results of user survey

The objective of part one was to confirm that our "assumed" parameters were the ones that were considered important when the service quality or service offers were evaluated. People were asked to list parameters that were considered important and to provide a motivation. We counted how many persons did mention a specific aspect as having a high degree of importance. The following aspects were mentioned:

- Availability & coverage
- Data rate (speed) important

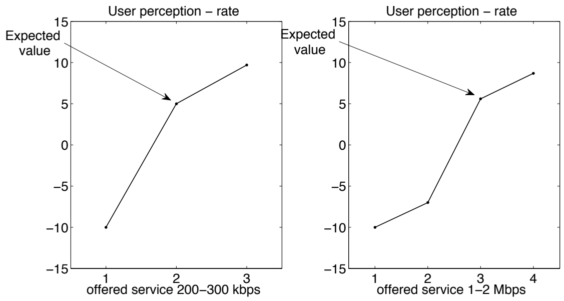


Figure 14. User perception & delivered data.

- Data rate not so important if availability is OK
- Security and reliability
- Price level
- Type of subscription
- User terminal & interface
- Ease of use

The results to a large extent confirmed our own assumptions of selection of “most important” parameters. Parameters considered to have “high importance” were “availability & coverage” (mentioned by 85%) and “price” (mentioned by 80%). “Security & reliability”, “ease of use” and “type of subscription” was mentioned by 50-60% of the participants.

Half of the persons mentioned “data rate” as being important, but it is interesting to note that equally many answered that data rate “is not so important” provided that the service availability is satisfactory.

The results of the second part on perception of service quality confirmed the findings by Twersky & Kahneman [25] when it comes to the user experience as being related to an expected value and also the “shape” of the “user happiness” function in Figure 3. Less than expected data rate resulted in a quick drop of perceived experience whereas an increase resulted in a much lower increase of the perceived experience. This is illustrated in Figure 14.

As an example consider the case where a user has a General Packet Radio Service (GPRS)/3G/HSDPA card in the laptop. If the user expects an ordinary UMTS connection then a 200 kbps data rate may correspond to the expected service. If HSDPA is available providing a 2 Mbps data rate the user probably not will be 10 times happier, but still happier. If HSDPA is available but the delivered data rate is 200 kbps the user experience will be much lower, all depending on the expectation of the user. In the same way if only a GPRS connection is available with e.g. 20 kbps data rate the user will get even more disappointed.

Also for the service availability the perceived experience decreased quite rapidly with a lower availability, see Figure 15 for average values. It is interesting to note that for individuals the transition from “good” to “very bad” was more rapid than indicated by the average values shown in the figure. When the availability went under some (personal) threshold then the rating in most cases rapidly went down to the lowest level (-10). This result is well in line with the results in the Ofcom results discussed previously indicating very high user value for coverage and availability.

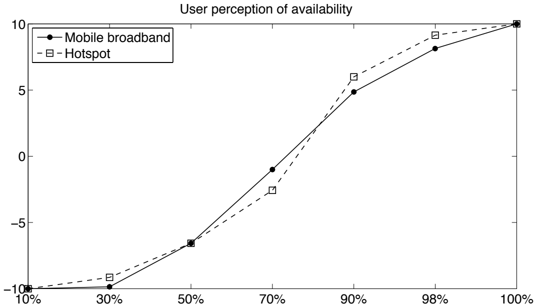


Figure 15. User perception & availability.

When it comes to the third part of the survey, to rate “the attractiveness” of different service offers, no clear pattern could be observed in decision making and the trade-off analysis. It may be that the sample size was too small. It turned out that the group “students” frequently used WLAN (for free) but not 3G (where payment was needed). People in the other group either had 3G subscriptions (often through their employer) or did not use wireless broadband at all. However, price seems to be the single most important parameter. The offers with lowest price got ratings between +5 and +10 and the offers with highest prices got ratings between -10 and 0.

7.2. Estimation of User Satisfaction Index

The average per user USI measurements with 300 MNs using the first P value weight set ($[-1, 1, 1, 1]$) are presented in Table 6 including the number of connected MNs and the normalized and absolute USI values. The USI value normalization is done so that the user without any period of disconnection has the normalized USI value 100 corresponding the absolute USI value of 1200 ($= 1200 * P4\ weight$). In the same way, a user disconnected all the time would have the value -100 corresponding the absolute USI value of -1200 ($= 1200 * P1\ weight$).

These measurements support the technical results showed in Section 6. Less time being in disconnected mode as showed in Figure 12 and increased service availability as presented in Figure 11 naturally affect and increase the user happiness as it can be seen from the USI values.

In practice, the first P value weight set $[-1, 1, 1, 1]$ represents the user behavior where the user is always equally

	Connected MNs	USI	Norm. USI
Legacy - No Coop	51%	21	2
Terminal - No Coop	57%	158	13
Network - No Coop	61%	255	22
Legacy - Coop	73%	569	47
Terminal - Coop	81%	733	61
Network - Coop	91%	988	82

Table 6. Connection and USI statistics with the first weight set.

	Connected MNs	USI	Norm. USI
Legacy - No Coop	51%	-126	-8
Terminal - No Coop	57%	37	2
Network - No Coop	61%	150	9
Legacy - Coop	73%	740	45
Terminal - Coop	81%	949	57
Network - Coop	91%	1227	74

Table 7. Connection and USI statistics with the second weight set.

happy whenever connected, i.e., different quality levels are not modeled. The negative value of $P1$ results in a high impact of disconnection.

Table 7 shows the USI measurements for the second P value weight set ($[-1, 0.25, 1, 1.4]$). The USI value normalization is done in the same way as for the first weight set with the exception that now the value range of the absolute USI value is [-1200,1650]. The second P value weight set results in different USI values since now different quality levels are distinguished, i.e., it does matter "how a user is connected". The negative weight value of $P1$ (disconnection) and a relatively low $P2$ weight value result in a negative USI value for the legacy case when the cooperation is not supported. When the cooperation is included, the situation gets better as indicated by a higher USI value; -126 vs. 740.

Both distributed algorithms perform better than the legacy one, but their performance is also relatively poor without the cooperation. This was expected, since without the cooperation both the access and service resources are limited. The *Network Centric* algorithm outperforms the *Terminal Centric* one but the difference is smaller than in the case of the first P value weight set, because in this case the P value distribution does matter.

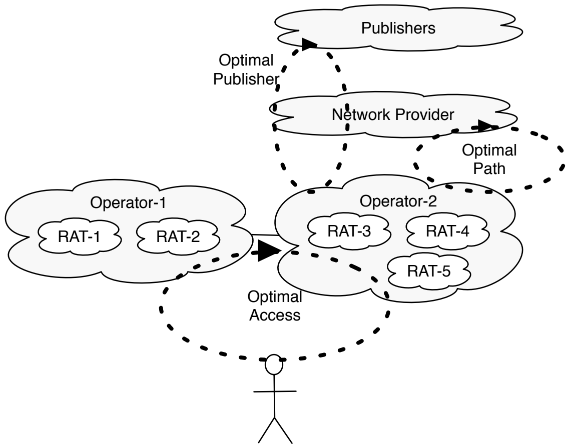


Figure 16. Multi-operator network scenario.

8. Extensions

The *Network Centric* and *Terminal Centric* algorithms were also considering end-users' communication needs like the requested service type, as the service availability were possible to consider through configuration of policies and profiles in the MRRMs, see Figure 4. This is especially an important aspect when we cannot assume that all services are equally reachable through all provided accesses. In addition to this feature, we discuss in this section future challenges of a decision making system while moving towards an *information centric* networking. Thus the *information centric* networking shifts the focus from looking at network as connected host towards a network connecting information producers with consumers. Related *information centric* work is research by Van Jacobsen [46] and yet another vision for a *information centric* networking is provided in [47].

One key feature is the introduction of caches in the network, data can be also stored at network nodes not only hosts. This leads to the situation where the same information can be available at multiple locations. The decision making system should with the *information centric* networking extended the access and services focus to also include content and delivery aspects, see Figure 16. In practise, this could mean for instance that the availability and location of temporary content storages ("caches") are also taken into account while evaluating and selecting available accesses. It is evident that the overall decision making system needs a common objective and the overall objective in the *information centric* networking is the performance of the application interaction Inter Process Communication (IPC) between different devices. This objective can utilize the set of different decision making subsystems (publisher, path and attachment subsystems) and the overall decision algorithm can apply them in different procedural combinations. How the actual algorithm and subsystems should be combined is for further studies.

The selection of service and data source includes the capability to determinate where the suitable candidates reside, i.e. the location of the service/data. The selection process should in normal case only require a network resolution function (source location selection). The resolution phase is clearly challenging as the future network integrates multiple caches in the network and the location selection is believed to have many identical data and service located in many places in the network. Here the 'closest' location typically is the best selection, however the 'closest' match can be challenging to understand as user and provider objectives and algorithm structure can vary.

It has to be clear that subsystem selection (subsystem optimization) can be contradictory compared to the overall performance. However the general method used in the described decision making is to include subsystem through weights for parameters and subsystems as contention can be addressed in a deterministic and fair way. On an algorithm level is the subsystem design clearly easier said than implemented, however the goal is to design a self-adaptable decision making system, which could be implemented for instance using both algorithm and constraint weights and adjust those when needed in order to change the decision making system's emphasis.

SLAs used between operators and service providers represent a fairly static nature of the network configuration whereas content caches are dynamic. This inherently implies that the decision making system has different timing phases and reaction times. The proper timing consideration should be addressed in the overall decision making. The timing properties can be handled such that faster decision loops contain more static parameters, e.g. faster decision loop includes more static selection criteria. The service and data delivery is handled through the path subsystem, however the path characteristics can change during data and service delivery. This is similar to the fast changing characteristics of the attachment subsystem like when the physical radio condition changes. The timing order of selection criteria updates for the overall decision making system is:

- Attachment criteria (fastest updates)
- Path criteria (faster updates)
- Publish criteria (fast and slow updates)

The future work will focus on the subsystem design of the optimal publisher and optimal path algorithms and how they should be interacting with the access selection process described in this paper. Some observations, the access selection is naturally limited to the geographical area of the user and the somewhat limited number of accesses available. On the other hand, the design of path and publisher criteria and algorithms are not limited to a geographical area and to design a scalable solution can be hard to achieve. The delivery based on the path subsystem is well known for the potentially NP complete problem alas it is impossible to find an optimal solution. These challenges will be further addressed in the 4WARD project [48].

9. Conclusions

We have shown that network cooperation, e.g. based on the *Network Composition* framework, has an essential role when designing a distributed decision making algorithm. Such algorithm is beneficial for the overall user experience when users are able to roam between access networks belonging to different providers. The way two provider networks like to compose is indeed a matter of business relations and trust between operators. Nevertheless the way composition is being performed also has an impact on the performance of the access selection. This should generally be taken into account when determining the wanted user experience. As the HO statistics presented in Section 6 indicates, the performance of a distributed decision making algorithm is also based on means to support HOs between RATs and operators indicating the importance of having powerful and flexible tools to support the network cooperation.

The general indication is that the additional coverage and supported services achieved through the network cooperation will increase the amount of potential customers that can be connected. Also, the attached customers would be more satisfied when their connections are more stable. This is the result of being able to freely select access network according to a richer set of constraints including both end-user and network preferences.

In general, the two new decision making algorithms worked as expected and resulted in network performance improvements. When the network traffic is close to or exceeding the congestion border, the algorithms are able to better exploit the available network resources than the legacy algorithm. Two simulation experiments resulted in different kinds of technical benefits due to their different simulation settings. The results in network cooperation simulations (multi-access environment) showed that the *Network Centric* and the *Terminal Centric* algorithms outperformed the legacy one in all measured technical metrics. Because these new algorithms were better able to exploit the network cooperation, the gap between them and the legacy algorithm was even bigger when the network cooperation was present. The results in competitive multi-operator simulations indicate that the power of the *Network Centric* and the *Terminal Centric* is in their capability of balancing the load between the available RATs in case of congestion, whereas in a single-radio case such option is not possible. It is also noticeable how well these two new algorithms were able to perform under a heavy network load compared to the legacy algorithm. When the mobile nodes' requested network capacity was in the range of 65%-110% of the maximum capacity, these new algorithm were still able to maintain approximately 30% higher number of connected users. This shows clearly how well these algorithms scale compared to the legacy one.

We have proposed and illustrated the use of a methodology to model and analyze the user experience of connectivity services. A main part in the analysis is the proposed performance metric called User Satisfaction Index (USI) which provides a mapping of the value of service parameters (e.g. data rate, availability) onto user experience. With different types of weights used in the mapping different types of services and different types of user perception can be modeled.

We have conducted a user survey on connectivity service parameters and user perception of services. The results support our choice of decision making parameters and are also in line with the parameter selection in Ofcom analysis on customer experience. The survey also provides useful input for selecting what kind of weight sets are used in the USI model and measurements.

The service availability and quality is related to the "short-term" (e.g. for each application session) user experience. Customer support and pricing have an impact on the long customer satisfaction. The former factor is not included in our study, but the latter is and the USI model covers it on a short-term basis. The current development with monthly flat rate subscriptions implies that the USI modeling and analysis will be of interest mainly for user experience of service availability, reliability and quality.

The technical simulation and USI results support each other. In a multi-access environment, the network cooperation results in gains for all evaluated algorithms indicating also better scalability. Clear benefits can be identified both for providers and for users, the overall traffic increases and the number of disconnected users decreases. As the USI results in Section 7 shows the type of application and the used algorithm affect on how the gained technical benefits translates into additional user satisfaction. When the used application is not quality sensitive, higher normalized USI values were achieved.

During recent years, the payments from international roaming have been one of the best source of profit for the mobile operators. However the situation is changing, when "wild west" style roaming pricing is no longer unheeded by the European Commission and upper bound for the roaming payments have been set in Europe. This decision will cut roaming profit considerably and is likely to drive operator towards new business cases and models. The problem in the essence in the new situation is how to cut operational costs from the operator-to-operator traffic. One solution has been growing in size, expand the coverage geographically and that way avoid the situation in general. But growing in size has its limits.

Alternatively, network and service operators in Internet have been fighting similar problems already years, while trying to minimize their transit costs. Internet's way of solving the issue has been establishing direct peering and sibling links between operator networks where applicable and where both sides of the agreement have seen the benefit. One of the most successful "peerer" worldwide is Google, who has enabled very vast low transit cost network through peering agreements with non-Tier-1 operators. In the middle of Fixed Mobile Convergence, maybe this suggests that national and local roaming (e.g. peering between operators) is something to be taken under serious consideration also in the mobile networking world and the concepts represented in this article are supporting this business evolution option.

10. EU disclaimer

This paper describes work undertaken in the Ambient Networks project, which was part of the EU's Information Society Technologies (IST) programme. In total, 41 organizations from Europe, Canada, Australia and Japan were involved in this project. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the Ambient Networks project. All information in this document is provided "as is" and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability. For the avoidance of all doubts, the European Commission has no liability in respect of this document, which is merely representing the authors view.

Some future extension work has been carried out in the framework of the IST 7th Framework Programme Integrated Project 4WARD, which is partially funded by the Commission of the European Union. The views expressed in this paper are solely those of the authors and do not necessarily represent the views of their employers, the 4WARD project, or the Commission of the European Union.

References

[1] P. Poyhonen and et al. Analysis of User Experience of Access Selection in Multi-Operator Environments. *The Third International Conference on Systems and Networks Communications (ICSNC 2008)*, 2008.

[2] 3GPP TS 23.251. Network Sharing; Architecture and Functional Description, 2004.

[3] 3GPP TR 23.234. 3GPP - WLAN Interworking; System description, 2004.

[4] M. Johnsson and et al. Final System Description - Public deliverable D18-A.4. Technical report, Ambient Networks project, 2008.

[5] P. Poyhonen and et al. Business Implications of Composition Framework in Ambient Networks. *Helsinki Mobility Roundtable*, 2006.

[6] J. Markendahl and et al. Ambient networking and related business concepts as support for regulatory initiatives and competition. *5th Conference on Telecommunication Techno-Economics (CTTE2006)*, 2006.

[7] O. Rietkerk and et al. Business roles enabling access for anyone to any network and service with Ambient Networks. *Helsinki Mobility Roundtable*, 2006.

[8] L. Ho and et al. Business Aspects of Advertising and Discovery concepts in Ambient Networks. *The 17th Annual IEEE Int. Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC06)*, 2006.

[9] P. Poyhonen and et al. Impact of operator cooperation on traffic load distribution and user experience in Ambient Networks business scenarios. *Global Mobility Roundtable*, 2007.

[10] J. Markendahl and et al. Analysis of operator options to reduce the impact of the revenue gap caused by flat rate mobile broadband subscription. *submitted to the 8th Conference on Telecommunication Techno-Economics (CTTE2009)*, 2009.

[11] J. Markendahl and et al. Systems Evaluation Results - Public deliverable D27-H.5. Technical report, Ambient Networks project, 2007.

[12] O. Rietkerk and et al. Business Feasibility analysis - Public deliverable D14-A.5. Technical report, Ambient Networks project, 2007.

[13] http://www.ambient-networks.org.

[14] N. Akhtar and et al. Impact of dynamic business relations and ”greedy” user behavior on business related signaling load in multi-provider networks with Ambient Network technology. *Global Mobility Roundtable*, 2007.

[15] J. Markendahl and et al. Analysis of Ambient Networks Mechanisms for Support of I-centric Communications. *WWRF #18*, 2007.

[16] P. Poyhonen and et al. Analysis of Network Cooperation in Terms of Operator and User Satisfaction. *The 7th Conference on Telecommunication Techno-Economics (CTTE2008)*, 2008.

[17] J. Markendahl and et al. Performance Metrics for Analysis of Operator Benefits of Network Cooperation in Multi-Operator Business Scenarios. *The 19th Annual IEEE Int. Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC’08)*, 2008.

[18] J. Andres-Colas and et al. Design of Composition Framework - Public deliverable D3-G.1. Technical report, Ambient Networks project, 2006.

[19] M. Prytz and et al. Multi-Access and ARI, Design and Initial Specification - Public deliverable D02-C.1. Technical report, Ambient Networks project, 2006.

[20] http://cordis.europa.eu/fp5/home.html.

[21] http://www.everest-ist.upc.es/.

[22] http://www.aroma-ist.upc.edu/.

[23] H. Tang and et al. Mobility Support: Design and Specification - Public deliverable D9-B.1. Technical report, Ambient Networks project, 2006.

[24] O. Pohjola and et al. Value-based methodology to analyze communication services. *CTTE 2006*, 2006.

[25] A. Tversky and et al. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*, 5:297323, 1992.

[26] J. Sachs and et al. Multiaccess management in heterogeneous networks. *Wireless Personal Communications: An International Journal*, 48:7–32, 2009.

[27] H. Mitomo and et al. A Behavioral Economic Interpretation of the Preference for Flat Rates: A Case of Post-Paid Mobile Phone Services. *18th European Regional ITS Conference, International Telecommunications Society*, 2007.

[28] A. Lambrecht and et al. Paying too much and being happy about it: Existence, Causes, and consequences of tariff-choice biases. *Journal of Marketing Research*, XLIII, 2006.

[29] R. Edell and et al. Providing Internet Access: What we learn from the INDEX Trial. *IEEE Networks*, 1999.

[30] R. Edell and et al. INDEX: A platform for determining how people value the quality of their Internet Access - project report #98-010P. Technical report, INDEX project, 1998.

[31] J. Gozlvez and et al. User QoS-based Multi-Channel Assignment Schemes under Multimedia Traffic Conditions. *Proc of the IEEE Conf. ISWCS 2007*, 2007.

[32] L. Badia and et al. Demand and pricing effects on the radio resource allocation of multimedia communication systems. *Globecom 2003*, 2003.

[33] P. Lindstedt and J. Burenius. *The Value Model: How to Master Product Development and Create Unrivalled Customer Value*. Nimba, Sweden, ISBN 9163063492, 9789163063497, 2003.

[34] J. Klang and et al. How to design and implement a global value management program to drive organic growth. *Value management symposium*, 2006.

[35] E. Sauerwein and et al. The Kano Model: How To Delight Your Customers. *Preprints Volume I of the IX. International Working Seminar on Production Economics*, pages 313–327, 1996.

[36] D. Walder. Kano's model for understanding customer-defined quality. *Center For Quality of Management Journal*, 39:65–69, 1993.

[37] J. Markendahl and et al. Operator Cooperation as a Competitive Advantage for Provisioning of Low Cost High Capacity Mobile Broad band Services. *ITS2008-Europe*, 2008.

[38] E. L. Farmer. Kano's Model presentation. Presented at the American Society for Quality; http://www.iems.ucf.edu/asq/presentations.htm, February 2008.

[39] P. Xavier and et al. Demand Side Analysis, Consumer Behaviour And Telecommunications Policy. *The 17th Biennal Conference of the ITS*, 2008.

[40] Ofcom (2006). Consumer Experience Research - Annex 4; Consumer Decision-Making in the Telecoms Market. Report on research findings, research annex, Ofcom, 2006.

[41] F. Kalleitner and et al. Mobility support: System specification, implementation and evaluation - Public deliverable D20-B.2. Technical report, Ambient Networks project, 2007.

[42] F. Kalleitner and et al. Annex to Mobility support: System specification, implementation and evaluation - Public deliverable D20-B.2. Technical report, Ambient Networks project, 2007.

[43] P. Poyhonen and et al. Analysis of Load Dependency of Handover Strategies in Mobile Multiaccess Ambient Networks. *Proc. of the Second Workshop on multiMedia Applications over Wireless Networks (MediaWiN 2007)*, pages 15–20, 2007.

[44] D. Hollos and et al. A study of handover strategies for mobile multiaccess Ambient Networks. *Proc. of the 16th IST Mobile and Wireless Communications Summit*, pages 1–5, 2007.

[45] I. Kadayif and et al. An integer linear programming-based tool for wireless sensor networks. *J. Parallel Distrib. Comput.*, 65:247–260, 2005.

[46] V. Jacobson and et al. Content-centric networking - White paper. Technical report, Palo Alto Research Center, 2007.

[47] B. Ohlman and et al. 4WARD Architecture and Design for the Future Internet - Public deliverable D-6.1. Technical report, 4WARD project, 2008.

[48] http://www.4ward-project.eu/.

Working Globally via Wikis while Innovating and Acting Together: Case Wiki-Based Knowledge Sharing Portal

Helena Suvinen
Agora Center
University of Jyväskylä
Jyväskylä, Finland
Helena.suvinen@jyu.fi

Pertti Saariluoma
Computer Science and Information Systems
University of Jyväskylä
Jyväskylä, Finland
ps@jyu.fi

***Abstract*—Wiki-technology is one of the new Internet collaborative service platforms that allows, among other things, building innovation networks for industries and research groups. To make these wiki tools really effective it is essential to understand human interaction problems within information and communication technology (ICT) based on scientific psychological grounds, rather than resorting to folk psychological intuitions.**

The problem with wikis is that their inbuilt interaction problems may substantially interfere with the development of open innovation communities. This is why it is essential to investigate psychological factors evident in the interactions within open innovation systems. This study, realized with a wiki-based portal, aimed to help build innovation systems within a particular industrial area. Our results identified several problems and point to a variety of user psychology reasons behind these problems. These reasons were organized into a classification system of problems experienced by users in task execution.

Keywords—Innovation, networked co-creation, wiki, psychology, usability

I. INTRODUCTION

User psychology refers to the psychological analysis of users in interaction situations [27], [29], [30], [35]. Psychological concepts, methods, and theories can thus be used to analyze human–technology interaction. This type of approach is required to find scientifically grounded solutions to interaction and design challenges. Thus, psychological knowledge is more useful than folk psychology intuition.

The importance of psychological knowledge has been known for decades, and it has been applied in various forms to human–technology interaction. Human factors, cognitive ergonomics, and psychological usability research are examples of good approaches in which psychological knowledge has played an important role [34]. User psychology collects these scientific themes under a single approach. It is characterized by the pursuit of psychological explanations to various interaction phenomena, and, in the long run, a desire to replace folk psychological practices in interaction design [3], [32], [36]. User psychology originates, to a great extent, from cognitive modeling, which also has the goal of applying psychological theories for analyzing interaction problems [1], [3]. However, user psychology is not restricted to cognitive psychology only.

One reason for the pursuit to build on scientific psychology is the constant developments in technology, which makes some interaction problems increasingly complex. Ubiquitous, pervasive, and embedded computing, as well as novelties such as WEB 2.0 or agent technologies, are providing a more challenging field for designing interaction processes than did the traditional keyboard and screen interaction types [34]. The crucial difference is that, instead of immediate usability properties, today's designers must pay attention to more holistic interaction processes. One must be able to thoroughly grasp what people want to do, why they want to do that, can they do what they like, and how they feel about doing what they want to do. And, because these are psychological questions, it is important to investigate, in different types of environments, what kinds of psychological categories are important in explaining interaction problems. As a result, it is possible over time to implement psychological knowledge in design processes.

The European Union (EU) has launched a number of European Technology Platforms for interconnecting industrial and research communities. This kind of knowledge-oriented policy (KOP) underpins the construction of a cognitive web that allows various communities to communicate successfully among themselves and thus acquire and assimilate essential knowledge [9]. It also promotes "hybrid forums" [7] that bring together, in an innovative way, the insights emanating from markets, companies, and research communities. A network of agents interacting in a specific economic industrial area and working within a particular institutional research infrastructure [7] provides a rich web of channels, with the advantage of high source credibility. Experiences and ideas can arise within the network [5]. An innovation system can be used to correlate and communicate knowledge [23] and to coordinate access to complementary knowledge. The Finish Fores Cluster Research portal, introduced at the Third International Conference on Internet and Web Applications and Services [41], is in focus of this study. Built on Mediawiki, the portal is a good platform for networked communication, co-creation, and global innovation. The development of WEB 2.0 opens up new possibilities for improved innovation management.

Wikis are groupware tools with collaborative capabilities. They work well in areas in which knowledge may be changing dynamically or where viewpoints differ about that knowledge and how to capture the informal knowledge that draws on the contributions [31] of a larger society for a specific domain. In organizations, the target area of wikis mainly consists of ad hoc problems in a distributed knowledge environment [44], [18]. Wikis can be used as a tool for continuous learning within and between organizations [18]. These tools can be private or public. The rapid growth of wikis, thanks mainly to voluntary contributors, shows that this environment as a service tool on the Internet merits serious attention [46], [47].

Wikis allow collaborative authoring in the context of a hypertext document set [33]. The main “wikinomics” principles are openness, peer-to-peer collaboration, sharing, and interacting globally [42]. This facilitates a real possibility for users to broaden their knowledge about a domain by openly sharing their own expertise and absorbing information from the large, global knowledge pools constructed via these wikis. Because the participants can possess totally different backgrounds regarding their educations and professional careers, this platform for combining of a variety of knowledge from several areas provides new opportunities for radical innovations to emerge.

Our article is presented as follows. In Section II, we describe our experiment: the materials, participants, procedure, and design. Section III contains the results of the research, with three qualitative examples of subjects executing the tasks, the number of errors the test subjects encountered, as well as the qualitative analysis of the errors the test subjects committed. The Discussion section delineates the user psychological problems behind these errors, the technical classification of the problems, and suggestions for developers. Finally, Section IV, Conclusions and Further Work, presents our proposals for remediation concerning the innovative Mediawiki-type knowledge-sharing portals. We end that section with our objectives for further work. Acknowledgments conclude the paper.

II. EXPERIMENT

We focused on Mediawiki as a portal developed for building *innovation systems*, particularly in industrial areas. There are several reasons why we chose this Mediawiki-based portal for our research. First, it has been the official tool since early 2008 for the Finnish Forest Cluster [11], which was established in September 2007 in line with the EU’s European Technology Platforms [10], designed to encourage collaboration among industrial domains and public and non-public research communities toward joint innovation. This particular portal was developed according to the Strategic Research Agenda of the European Forest Sector within the European Technology Platform [39]. It became obvious that efficient tools such as portals are needed to enable researchers from traditional and emerging areas to contribute to national and Europe-wide research agendas.

By use of the portal, research groups can demonstrate their competencies, post research ideas, and plan projects. If needed, they can get help from portal facilitators familiar with the domain who champion research needs derived from the research agenda and the program related to it. These kinds of programs have been launched by Finland’s Strategic Centre for Science, Technology and Innovation (TEKES). First of these programs, the Finish Forest Cluster (now, Forestcluster Oy), was established 2007 and our studies focused on its research portal.

The study dealt with the usability of a Mediawiki-based portal, with the aim to increase awareness about the technical usability of this kind of application [41].

A. *Materials*

The target in this study was a Mediawiki-based platform [49] written with the PHP scripting language for a Linux operating system. Mediawiki uses an Apache web server and a MySQL database. The idea behind this application was to gather the knowledge of various researchers and their teams, as well as other actors in this forest domain, into a single virtual location, allowing all participants to build up their knowledge, to share it with others, and employ it. The application was built for collaborative communication, co-creation, and open innovation in a networked environment.

The tests were recorded by way of an Easy Screen Recorder, which documents user behavior on the platform and the mouse clicks on the screen, as well as the speech of the test subject.

B. *Participants*

The experiment involved 14 researchers as test subjects. The test subjects comprised 7 women and 7 men, aged 18-48 years. All were very experienced computer users and information seekers: two of them had between 5 and 10 years of computer experience and the other 12 had used computers in their work for more than 10 years. Of this test group, 12 searched for work-related information daily, and the others a few times a week during work hours; all of them searched to some extent in their spare time. All of them also had used Wikipedia in the past, 13 of them when searching for information. One had built his own wikis and inserted information into them. Thus, the subjects demonstrated a deep understanding of the use of computers and Websites, mostly for work but also for personal interests. In addition, the subjects were motivated information seekers via the Web since they were postgraduates of various information science disciplines. These statistics are described in Table 1.

TABLE I. THE BACKGROUND INFORMATION ON EXPERIMENTAL SUBJECTS

N = 14	Experience/years	Sum
Use of computers	5-10 years	2
	>10 years	12
Use of WWW sites	6-7 days/wk	11
	3-5 days/wk	3
Information searching at job	Daily	12
	Few times/wk	2
Information searching on free time	Daily	5
	Few times/wk	8
	Few times/month	1
Use of Wikipedia		14
Use of Wikipedia for	Searching info	13
	Adding info	1

The test subjects were asked for their principal reasons for using Web sites. The foremost reason was for work-related information, followed by studying, and then for commercial use. Hobbies and entertainment were equal reasons. The least important reason for the subjects was use of public services.

C. Procedure and design

The research data were gathered in experiments where one test subject at a time navigated through the portal while thinking aloud about solving four different tasks. The navigation from an entry page to the target page could be executed via different routes but the most efficient way was known as the *optimal path* [12], [2]. This shortest path consisted of the Web pages the user had to visit, and there can be “several optimal paths for one information search [21].” The tasks studied and their optimal paths are presented here.

1) Find the main idea behind this Mediawiki-based portal.

Optimal path: Read from the Main Page.

2) Find the available research groups.

Optimal path: Main Page→ Click one of the eight Research Communities names→ Click some Research Group name→ Read the name of some researcher.

3) Find how to add wiki-type information into this Mediawiki-based portal.

Optimal path (When no information on that topic is available yet): Type some word in the Search box→ Press the GO button→ Click the Create This Page link→ Add information.

Optimal path (When information on that topic is already available): Type a word in the Search box→ Press the GO button→ Click Edit→ Add information.

4) Find how to format and organize wiki-type information in this Mediawiki-based portal.

Optimal path: Main Page→ Click the Quick Guide in the left bar→ Scroll to the end of the page and read the instructions→ and understand content.

The experiment was conducted in December 2006 at the University of Jyväskylä’s User Psychology Laboratory. During the recording researchers observed the situation and the behavior of the test subjects. Following the tests, the data were analyzed and organized to investigate the usability of the Mediawiki-based interface and offer suggestions to its developers.

III. RESULTS

In this study the results were analyzed first on the basis of success (the number of subjects able to succeed executing the task) and then on the basis of using the optimal path (the number of subjects able to execute the tasks on the basis of the optimal path). The final analysis was based on the kinds of difficulties/problems the test subjects had during tasks executions.

A. Three qualitative examples

While it is not possible to present all our analyses in detail, for want of space, the examples below will give the reader a concrete view to the navigation processes and to the subsequent analyses. Our examples involve three subjects: one each performing Task 3 (inserting information to the portal) or Task 4 (editing text in the portal), and the final subject performing all four tasks. These examples are described here step by step in a text form and illustrated in figures as well. The Easy Screen Recorder captured the subject’s navigation decisions in addressing the stated task. Their verbatim comments are stated in italic in quotation marks, and paraphrased comments are present in parentheses. Although they were asked to think aloud, there are silent phases in their task executions and this is represented with a dash (—) in verbatim quotes. Editorial comments are in brackets. The portal text at the time of the study was only English and thus subjects were Finnish non-native speakers of English; the direct quotes are translations by the researcher.

The first one shows how troublesome wikis can be for new users, regardless of whether they are experienced computer users or information seekers on the Web. In the second example, the test subject comes through easily, giving then her own contribution to the developers of this particular portal. In the last example the user is not familiar with wikis and, being an excellent computer user, performs quite well with this particular portal under study.

1) Subject 1, Task 3: Find how to add wiki-type information in this Mediawiki based portal.

Main page→ clicked the Smart Products link→ scrolled down→ clicked the Research Group link→ clicked the Back button→ clicked the Upload File button→ clicked the Browse button (wondered where the information would be saved)→ clicked the Cancel button→ scrolled down→ clicked the Community link→ clicked the first article under the letter C→ clicked the Back button→ clicked the first article under letter N→ clicked the Ideas for Future Research Projects link→ clicked the Ideas link (wondered whether it would be possible to edit here)→ clicked the Edit link at the top of the page→ clicked the Back button→ clicked the History button→ clicked the Back button→ clicked the

Move button at the top of the page→ clicked the Back button→ clicked the Unwatch tab→ clicked the Discussion button→ clicked the Back button→ clicked the Back button→ clicked the Upload file link→ clicked the Back button→ clicked the Back button→ clicked the Main Page link→ clicked the Back button→ clicked the Community Portal link→ scrolled down the Community Portal page→ clicked on the Selected Research Communities link→ scrolled the page→ clicked the Help link in the left bar→ clicked the Wikipedia Edit Page link [test subject got to the Wikipedia page and did not even notice that she had left the portal under study]→ clicked the Back button→ clicked the Back button→ clicked the Recent changes link on the left bar→ clicked the Community Portal link on the left bar→ scrolled down (admitted not knowing how to add information for this site)→ clicked the Special Pages link [while the researcher were advising her to use the Help page] → clicked the Back button [researcher advised her to select a word for the search box]→ typed *usability*→ clicked the Go button→ [was advised to move to the Create a page link on the opened page]→ (understood the procedure).

In Figure 1 the Optimal Path in Task 3 is illustrated in white circles. The large number of steps the Subject 1 took while trying to insert information to the Mediawiki–based portal are in black ovals.

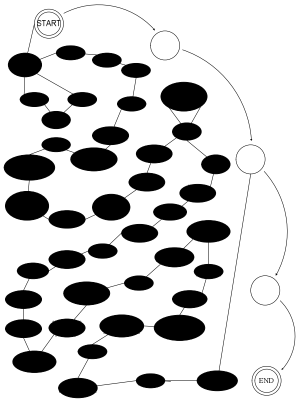


Figure 1. The results of the Subject 1executing Task 3.

This example was about inserting information. Only four test subjects were able to execute this task independently and none of them did so by following the optimal path. One subject vocalized her ideas regarding the application and about possible remediation of problems while she was searching for guidance in the portal in order to edit information. She managed to perform the task, using 4 minutes for five steps, but not along the optimal path.

2)Subject 5, Task 4: Find how to format and organize wiki-type information in this Mediawiki based portal.

"Here is the HTML editor," clicked Help→ scrolled down the page [when reached the right place] *"I assume that Wikipedia does the overall formatting, but I guess that is in the User's Guide.*"→ scrolled to the actual text about content formatting at the bottom of the Help page [thought she was in the User's Guide]→ clicked the Back button [in the interface and got back to the right Editing "Digimedia" page]→ filled in information on the page→ started formatting with the buttons at the top of the page, "*This has its own, special syntax for the formatting that differs in a very interesting way from everything else I am used to.*"→ clicked the Forward button in the interface, "*This is not HTML coding. This is some kind of totally individual marking language and text formatting. Obviously this is specified in the instructions, this syntax, which does not make it easier to insert content. So one would have to study a new content formatting language—. Add an asterisk. The more asterisks, the deeper the level—. In Word, the formatting is hidden from the writer. There is WYSIWYG—. In this we go back to the 1980s, when the formatting was done by coding separately with different symbols—with some code language. And this code is, for me, totally new. I have never seen anything like this—. This might come from some formatting language that I don't know—. One way to improve this would be to make it consistent with HTML code—so those who know HTML formatting, they would not have to learn a new code—. Does this come from the platform* [Mediawiki]*? —There is always the threshold of learning new* [tool]*—. All wikis should be the same—. These transitional periods for these kinds of new systems and environments are always cumbersome to the users and frustrating. They think that it is needless to learn a new coding language again—.*" [Task completed.]

In Figure 2, the Optimal Path in Task 4 is illustrated in white circles. Subject 5 took only three extra steps, pictured in black ovals, to find out how to format the added information.

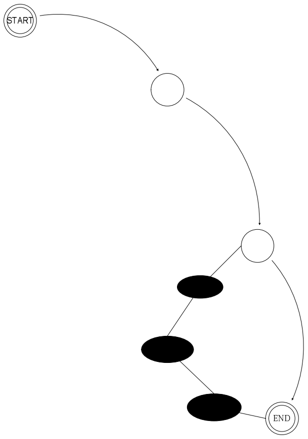


Figure 2. The results of the Subject 5 in executing Task 4.

Altogether 10 subjects managed to edit the information and three were able to do it following the optimal path. Only four subjects managed both the inserting and editing information to the portal and editing it (Tasks 3 and 4), but none executed these two most important tasks using the optimal paths.

3) Subject 13, Tasks 1- 4: Task 1, Find the main idea behind this Mediawiki-based portal.

Main Page→ clicked the Behind This link→ clicked the Main Page→ found the header What is New in Forest Cluster Portal [read]→ *"OK. It is on this page."*

In Figure 3, the Optimal Path in Task 1 is in white circles. It did not need any steps, just understanding that the information was on the Main Page. Still Subject 13 needed two additional steps (black ovals) for understanding this main idea.

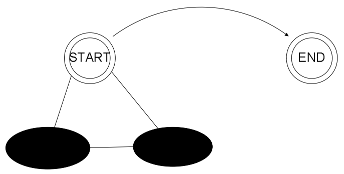


Figure 3. The results of the Subject 13 executing Task 1.

Task 2: Find the available research groups.

Main Page→ clicked the Community Portal link→ clicked the All Research Groups link→ clicked Template Research Groups→ clicked the Back button→ clicked the Groups: Scoma link→ *"OK. It is here."*

In Figure 4, the Optimal Path in Task 2 is illustrated in white circles. Subject 13 needed four additional steps (black ovals) to be able to read a researcher's name in one of the available research groups.

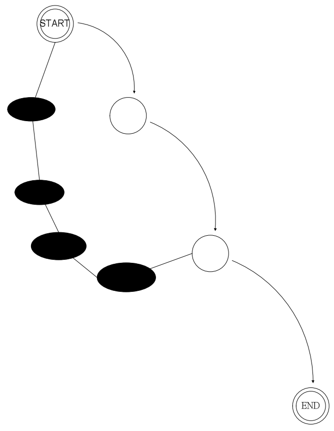


Figure 4. The results of the Subject 13 executing Task 2.

Task 3: Find how to add wiki-type information in this Mediawiki-based portal.

Main Page→ clicked the Others link→ clicked the Back to Main Page link→ clicked the Intelligent Resource link→ clicked the Main Page link→ clicked the Others link, *"I assume that my information would go to the Proposals or Ideas partition. No this does not so—."*→ clicked the Community Portal link→ clicked the Smart Fibre and Resources link→ clicked All Articles→ clicked the Proposals link→ clicked the Community Portal link→ clicked the All Categories link, *"It's a little bit confusing, flicking through these categories.— There are only 42—it's working quite badly…—. It is not sensible to show hundreds of available categories* [when only 42 results were available]*."*→ Clicked Main Page, *"I can create new categories, but how am I going to link them to other pages?"* [Researcher asks if the subject commonly uses the Help option.] *"I didn't expect that wikis would need help—; I thought it is easy to insert information in wikis!"*→ clicked Help→ clicked New Page Creating→ clicked Main Page→ typed *data mining* in the Search box→ pressed the Return key→ arrived at the Search Results page→ *"I found it—. It's same problem with the navigation as before. This shows there are up to 500 pages to see but there exists only five pages. This is confusing—. It is even possible to choose previous pages. It's a bad problem!"*→ opened a new working window→ typed "text mining"→ pressed the Return key→ opened the link to the Create This Page: Text Mining→ started editing the blank page Text Mining → "*OK. This is it."*

In Figure 5, the Optimal Path in Task 3 is in white circles. The additional steps the Subject 13 took while trying to insert information to the Mediawiki-based portal are shown in black ovals.

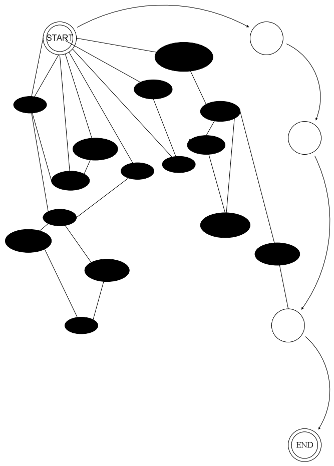


Figure 5. The results of the Subject 13 executing Task 3.

Task 4: Find how to format and organize wiki-type information in this Mediawiki-based portal.

"I suppose there are instructions for formatting in Help.→ pressed the Quick Guide link, *"Yes."* → scrolled to the end of the page→ *"OK,"* and wrote some sentences onto the Text Mining page. *"How do these icons work? — First we choose the text— ok— not all symbols are familiar to me—. Luckily, there are tool tips — makes it much clearer—. It would take some time to learn all these notations."* [The researcher asks if the subject thinks it is a good tool for formatting.] *"Enough for me—. Some would like to have WYSIWYG."* [Researcher asks if it would be easy to implement the WYSIWYG for this application.] *"Tools exist that generate HTML code—. Yes certainly—my opinion is that this wiki-formatting is generated so that you don't have to write HTML—. Yes this is a quite interesting way of formatting."*

Figure 6 shows only the optimal Path, which Subject 13 was able to follow correctly. He wrote some information on the page and formatted it instantly.

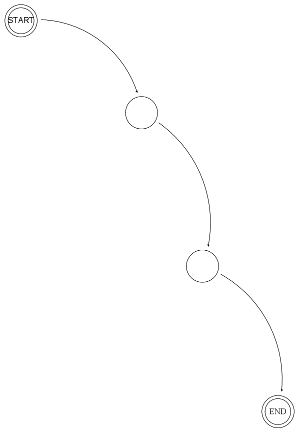


Figure 6. The results of the Subject 13 executing Task 4.

This particular test subject performed well with three tasks, even though he used the optimal path only in the fourth task. Only Task 3, adding information, was quite confusing for him. Even though the use of the wiki platform was quite transparent to him, he had to click around the portal while performing these four tasks, and once the researcher prompted him to use the Help. During the last task, while he was formatting the information he had typed in the portal, he gave some useful observations and ideas on how to evolve this kind of tool.

"Basically—clear layout— if someone has used Wikipedia before— maybe, if you should insert new information here—. Not very many would use Help—. There should be 'Welcome to insert information'—. Usually I can keep trying for a long time—.I don't give up easily—. Maybe it is more like— yes— in one way or another I insert something—. Maybe there does not exist a technology threshold—. You can always put in plain text— without any bolds or other formatting."

When they were inserting information, the users were confused between concepts such as article, page, and file, on the one hand, and wiki-type information, on the other. These differences are not obvious for new users to wikis. Younger people are quite likely to adapt into these working environments, judging by their habits of discussing and sharing their knowledge in the networked world, even in their spare time. This willingness to discuss and share knowledge is most likely going to change the way that work is conducted in organizations. Regarding older experts, capturing their vast knowledge in particular domains is not so simple. Equivocal concepts involved make their use of these cooperation tools more difficult for them.

B. Number of errors

The main quantitative results are presented in Figure 7. Columns 1-4 depict the tasks that the subjects were asked to do. All 14 of the subjects solved the first two tasks, 4 were able to add information (Task 3), and 10 were able to format that information (Task 4). The fifth column shows that only 4 subjects were able both to insert and to format information in the portal. Searching, inserting, and editing are critical functions with these knowledge tools. Properly designed, these functions prepare the way for collaboration, co-creation and innovating in networked societies.

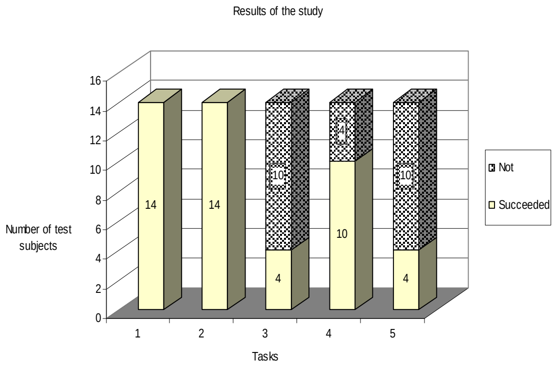


Figure 7. The results of the four tasks in the experiment with the Mediawiki based portal. Column 5 contains summary information showing the number of those who managed to do both tasks 3 and 4.

The optimal paths seemed more troublesome for the users. Six subjects managed this aspect in the first task, 4 in the second task, and 3 in the fourth task. In the third task, where the subjects were asked to insert information to the portal, no one followed the optimal path. These results are shown in Figure 8.

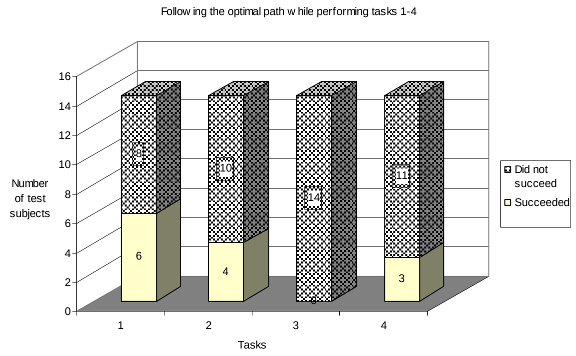


Figure 8. The results on how the subjects were able to perform the four tasks by way of the optimal paths.

The reasons for the problems that the users came across are explained in Subsection C. The number of users (NOC) who came across each problem is given and there is one example of each problem. The difficulties have been divided into several different categories.

C. Qualitative analysis of the errors committed by the test subjects

Quantitative distribution does not tell us much about the type of difficulties people had with this wiki. They merely show that certain types of task were difficult. It would be very important to also consider the quality of errors and their psychological interpretations. The latter indicates the kinds of psychological factors that can explain what is happening.

In order to get a better idea about the errors and be able to examine their psychological backgrounds it is essential first to discover the errors and then to find a scientific basis for understanding why these error generating points are difficult for people. This is why we present the qualitative analyses of all data classified into psychologically meaningful categories.

This type of categorization is a normal procedure in qualitative analyses [8]. However, using psychological categories presupposes interpretation of the observations in psychological terms. From a psychological point of view this process is in many respects methodologically close to clinical processes. In personality assessment, for example, psychologists categorize the symptoms and patients to connect the observed cases with psychological knowledge.

This is a rather detailed list but we think that it may be better from a developer's point of view than mere examples or numeric information would be. Nearly all errors we found in this study derived from the Mediawiki platform, but these same problems can be found in other ICT services too. Note that unlike earlier, verbatim texts are not italicized.

1) Difficulties in perceiving

We classify any failure as perceptually originated when the problems in navigating can be explained on the basis of some perceptual phenomenon that can be elaborated in perceptual terms.

Color of links: Subject 7 in Task 3 stated, "Is here some link? … It cannot even be seen that there is a link here. It is nearly of the same color [as the other text]." NOC 5.

Disconnecting the page by a picture: Subject 9 in Task 3 noted, "This picture disconnects the page [in the User's Guide]". NOC = 1.

Page resembling code: Subject 7 in Task 2 regarding the Recent Changes page, "Not a user friendly page. …Looks just like some code." NOC = 1.

Confusing namespaces area: Subject 2 in Task 3 noted, "Search in namespaces at the bottom deals with an area that is too wide." NOC = 2.

User's glance not directed to the right place: Subject 12 in Task 3 said, "This directs my glance to the middle of the page because there is the 'Munch' [the word she searched], not towards the top of it as it should. …I thought I should find it here nearby [word 'Munch'] and not on the top." NOC = 1.

Too much text: Subject 8 in Task 4 said, "Too much text to read.… Probably I would ask for advice from someone." NOC = 4.

No WYSIWYG: Subject 13 in Task 4 said, "For my needs, these editing icons are enough, but someone else might like to have WYSIWYG." NOC = 1.

2) Difficulties in understanding

Page layout leads to wrong navigation: Subject 3 in Task 2 saw that "Links to Wikipedia Help on top of the Help page guided users to Wikipedia." NOC = 1.

Hierarchy of the concepts not evident for users: Subject 5 in Task 2 had problems with the hierarchy of concepts. NOC = 1.

Confusing concepts: Subject 5 in Task 3 found, "The relation between Page, Article, Research Project, and Research Group is not obvious for the user." NOC = 15.

Confusing instructions: Subject 11 in Task 3 noted, "The Help is a little bit hard to understand because after [instructions of] creating a page, there is an instruction: 'However, this way isn't recommended.'" NOC = 1.

Confusing information on the page: Subject 9 in Task 3 went to the Recent Changes page and said, "No one is going to understand this page." NOC = 2.

Confusing messages, such as "There is no page titled [or article title and page text] *matches":* Subject 11 in Task 3 said, "The 'Usability' page already exists here.… This is confusing." The Search Results Page gave different matches for the page that the user was searching for, for the Page title, and for the content of the pages. NOC = 2.

Confusing content: Subject 1 in Task 2 said that the page content did not seem self-evident for her. NOC = 1.

The logic is not clear: For Subject 5 in Task 2, the logic of the portal was not that clear. NOC = 1.

Editing wrong page: Subject 3 in Task 4 edited the Help page. NOC = 7.

Hierarchy not evident for the user: Subject 9 in Task 4 noted, "It is easy to get lost in this interface." NOC = 5.

Information storing hierarchy not evident: Subject 7 in Task 3, queried, "How do I know, if I insert some information here, whether it goes to Wikipedia or to this portal?" NOC = 2.

3) Transfer and memory

Difference between menus on the top and side: Subject 12 in Task 4 noted, "With these top and side menus, it is somehow difficult to see where to start." NOC = 1.

Difference between a portal and wiki: Subject 7 in Task 3 said, "I don't understand what the difference is.— Why is the wiki here?" NOC = 1.

Difference between Help, User's Guide, and Quick Guide: Subject 5 in Task 2 noted, "The difference between User's Guide, Help, and Quick Help is not evident for the users." NOC = 5.

Difference between Go, Search, and Return: Subject 3 in Task 3 stated the difference between the Go/Search buttons and the Return key was not very clear to him. NOC = 3.

Confusing information architecture: Subject 5 in Task 2 said, "The way the information architecture is constructed in wiki type portals is new and not yet well known." NOC = 2.

Confusing namespace listing: Subject 11 in Task 3 wondered, "Search in namespaces—. I could select— but what is the idea in this—. Default is Main—. This is a little confusing." NOC = 6

New formatting: Subject 5 in Task 4 found it difficult because "I would have to learn a new formatting style—. Not even HTML code." NOC = 1.

No breadcrumb trail: Subject 3 in Task 4 said, "There should be a breadcrumb trail so that the user would know where he located." NOC = 2

No site map: Subject 10 in Task 3 commented, "I didn't find the Main Page of the dictionary." NOC = 2.

Confusion about location. Out of the portal and in Wikipedia: Subject 7 in Task 4 stated, "I don't even know whether I am in this portal or in Wikipedia." NOC = 5.

4) Motivation

Motivational reasons: Subject 4 in Task 4 saw that the usage motivation is dependent on the necessity to use the portal. One might use it only if it was necessary for work, but it was too complicated to use with hobbies. NOC = 4.

Users do not use Help: Subject 9 in Task 3 acknowledged about the Help page, "No one reads this kind of information." NOC = 5.

5) Other functions

No inspection for formatting: Subject 9 in Task 4 wondered why "this does not inspect the formatting after one has inserted information." NOC = 3.

Linking stored information between pages: Subject 13 in Task 3 asked, "If I insert something, how do I link it to other pages?" NOC = 1.

IV. Discussion

The experiment illustrates the substantial difficulties that the subjects can have with the two essential tasks in using these types of portal: inserting and formatting the information. These problems can be traced to the inherent problems in the Wiki platform. Qualitative investigation of stumbling blocks makes it clear that human psychological functioning is not sufficiently understood in the construction of wikis.

The data were gathered through the so-called thinking aloud method. When a test subject is quiet while working, the researcher is unsure whether the subject is reading the content or uncertain or confused about how to continue with the task. While observing the situation, the researcher is unable to perceive if there is a problem, any of the situations where he/she might be encountering a problem, or any of the reasons for the test subject's problems. When the subject thinks aloud, he/she provides the researcher a level of insight into what the subject is perceiving and reasons why he/she is completing the task in a particular way.

A. User psychological reasons behind the problems

Our aim is to categorize the usability problems raised by the subjects in psychological terms. This allows a direct connection between the problem points and the currently available psychological knowledge. Further, we can elaborate our conception of the human mind in interaction. One of our main goals is to develop psychologically grounded design principles. However, while modern engineering design is generally based on scientific knowledge, interaction design is mostly intuitive and based on folk psychology [36].

Some of the interaction problems were caused by perceptual difficulties. We found 15 such cases. One problem was caused by colors that were too similar and people could not easily discriminate between them. Properly used, color is a very good directive search cue, but if colors are too similar, the benefit is eliminated [43]. Situations comparable to this would be text too closely resembling code or simply too much text on a page. In both cases, discrimination of the target becomes problematic because the target and the background information confuse the user during a search [25], [40].

Another important problem is information invisibility, in which all the information necessary for controlling ongoing actions cannot be visually provided to the users. Essential information may be placed outside the screen. This of course can also be regarded as a memory problem, but because its correction is based on making missing information visible, we have classified it as a perceptual difficulty. We found three cases like this. Another demonstrative case of this effect is when important information is allocated outside the focus of the gaze. These problems often are due to overlong texts and incorrect page sizes. It may also be that WYSIWYG did not work adequately.

A third general concern about user psychological problems in wikis has to do with understanding. Understanding involves a human's ability to encode information into one's own mental representations. This means comprehending the meaning of information that is a word, a sentence, or an event.

In this experiment, subjects confronted many types of problems in understanding the interfaces. Some of the commands used in the portal were quite ambiguous and partly incomprehensible. Terms were not explained. Feedback was inadequate and thus prevented the subjects from understanding the operational logic of the system. Navigation tools were inadequate or absent. The system did not indicate where the user was at any given time. The purposes of actions on some pages were not explicated or intuitive. Thus, expectations were difficult to comprehend.

Understanding is a complex process. One has to be able to take in and comprehend meaningful details and determine the right way of doing things. It is essential to be able to follow simultaneously many different types of information flows [24], [37]. In this study, the user had to be able to control at the same time navigations and program control flow while not forgetting the actual innovation text as well. Failure in processing any single component of this process may cause the entire process to fail.

Fourth, some of the problems were related to transfer, in which earlier learning negatively or positively affects learning new information [14]. For example, link colors were non-standard or inconsistent, causing problems for subjects who were used to different color codes. The same was true with the wiki's original features.

Two major types of situations can affect transfer. First, external user culture may influence the usability of a new system. People have usability habits and practices. If a new system or program differs fundamentally from the previous versions or familiar application, it will cause problems of negative transfer. Second, the system itself may be inconsistent and thus cause incorrect expectations in users.

One important rule has been found in research into transfer: The more overlapping the features of two interfaces, the greater the positive transfer between them [38]. Transfer is thus a very important phenomenon to consider when addressing user–technology interaction problems.

Fifth, we found concerns related to motivation. In human–device interaction, success is one of the main motivation factors for users. In e-learning, for example, people who are able to solve interaction problems often seem to become proud and self-confident, while people who fail might lose their motivation and often demonstrate a negative attitude towards e-learning [16], [17].

The final major set of problem we found were clearly technical bugs that hindered usability. These will be addressed separately because their resolution cannot be based on psychological information.

We focused on psychological analyses of the difficulties people face when interacting with technologies because such analyses can indicate to advanced interaction specialists how problems could be solved. We can speak of explanatory frameworks, among them cognitive, emotional, and socio-cultural, when we refer to the theory languages we can use in solving an interaction problem. We simply have to find the right explanatory framework in each case to be able to explain the problem. This means the right psychological basis for explaining why the interaction does not work optimally.

B. *Technical classification of usability problems and suggestions to developers*

While we need look the problems from a psychological point of view, it is important to pay attention to the technical side of interaction as well in order to get a clear idea of the types of corrections needed. We have listed the technical problems in a table in Appendix 1.

The area of *concept clarification* contained two kinds of problems in this portal. There were concepts that were not self-evident, so users either did not understand their meaning, or misinformed. A Tooltip help, which explains the word in a wiki page and does not merely replicate it, would make it easier to understand the concepts.

In the area of *content facilitation,* a distinction should be made between content, its generation, and its use. Titles should be clear and short and be made up of highly informative words. The first two paragraphs of a Web page must state the most important information.

There are several possibilities for *function facilitation* in wiki sites that can be employed well. First, there should be a page map with the location of the user visible, as well as a complete path from the home page down through all the levels of the information architecture. Second, the difference between the links on the left and at the top of a page (in this particular Mediawiki-based portal), as well as at the top of an article, should be made more clear to the users. In addition, the difference between the Go and Search buttons should also be made clearer, as well as how the Return key works in this portal. Finally, the use of namespaces at the bottom of the search results page should, likewise, be explained better.

Regarding *page elucidation,* we had several recommendations. First, pages with multiple content areas (header, its design element, and content) were confusing. To eliminate confusion, the line below the header that was used as a design element should be above the header and content area, rather than between the header and content. Moreover, pages such as this that were in the portal (groups of headers, content and edit functions) was not self-explanatory. In addition, users do not like scrolling very much, and for this reason, too much information on the Help pages induces errors when users search the rules in order to add information. Finally, employing effective visualization would make it easier to understand the functions available in this particular portal.

Well-executed *function automation* can prevent many errors. An Add button, which opens a template for writing new information, would facilitate the process. In addition, function automation should attend to the details of processing the added information. There should be access to help online, with an option to jump to the specific answer on the Help page. Hovering the cursor over a link should give a Tooltip text with more information about the link.

The study site contained so much text that only highly motivated users would read it all. Providing in the portal a demo on how to use the port would make it easier to get a mental picture of the site. It is obvious that *training and motivating users* are big challenges for the designers of this portal. Therefore, the working cultures of open societies that are adapting wikis for particular industrial and research domains should be built or adapted through carefully planned user training. This would facilitate the emergence of truly open societies creating new knowledge and innovating together on a particular domain.

V. Conclusion and future work

Because the trend in emerging innovations is about firms operating in a coordinated manner in networks [22], the nature and necessity of collaboration within networked environments must be understood. In this context, large-scale collaborative development of tools, like wikis "and use of open source software merits a great deal of further attention and analysis" [48].

The role of human capital in innovation is important as well, both at the firm and the aggregate levels [28]. Wikis are tools that enable organizations to gather some of this capital for collective use, while the "open innovation paradigm assumes that there is bountiful supply of potential useful ideas outside the firm" [6]. This means "valuable ideas can come from inside or outside the company and can go to market from inside or outside the company as well" [6]. "Innovation co-operation requires active co-operation with other firms or public research institutions on innovation activities (and may include purchases of knowledge and technology)" [28].

Wikis can be bridges to sources of information, knowledge, technologies, practices, and human and financial resources. They can be used a link that "connects the innovating firm to other actors in the innovation system: government laboratories, universities, policy departments, regulators, competitors, suppliers and customers" [28]. Moreover, "the supply of knowledgeable minds to which innovating firms have access is perhaps the most crucial aspect of the innovation systems approach and of innovation policy for it is individuals within organizations who are the elemental components of innovation systems" [23].

Even with these kinds of knowledge-oriented policies, innovation systems, and global cooperation, our results illustrate that interaction through open innovation platforms is neither easy nor straightforward. Though the literature on psychological problems that users may encounter is solid, the principles are not easy to apply [3], [4], [13]. Consequently, people encountering problems may give up the idea of sharing their respective knowledge on a particular innovation domain and in interacting via continuous feedback to improve the creative principles [26].

To avoid this potentiality, some proposals for remediation are offered. First, it must be determined how new users can obtain the correct mental representation of a wiki: its structure, functions, and the activities the user can perform. These can be represented through demos, and with various training procedures, such as sandbox and illustrative sitemaps. Second, the content of a wiki must be clearly distinguished from its functions. Third, because contributing to a wiki is an entirely new working culture, employees need a kind of usability that is different from the usability of a common interface. The platform should evoke understanding of the very idea of knowledge sharing, co-creating, and contributing within an open society. To succeed at all of these challenges, the developers should have a deep understanding of human psychology.

These kinds of portals, and especially the possibilities they can offer for global networked co-creation and innovation, demand the ease of usability. For the portal in this study, the remediation we suggested is already in process. In the wider perspective, work and other activities in networked environments and open innovation with peer experts or other users already are creating a global working model that is growing fast. The users should be able to use these portals easily, without wasting time learning another new tool, new application, codes, or action model. They should be able to simply concentrate on the sharing their expertise of the subject as it evolves, thereby partaking in and adding to the knowledge of the societies involved.

Designers should rethink how user-created content [48], especially in open innovation [6] context, can best be used by people with low levels of computer knowledge. The current usability problems with wikis, as illustrated here, may compromise the very idea of open innovation. In this experiment, the subjects were postgraduate students of information systems and computer science, and thus may be looked upon as experienced users. Nevertheless, they still experienced problems with relatively elementary tasks on this platform. This shows that poor understanding of human psychological requirements can lead even quite sophisticated users into problems. This runs counter to the very philosophy of open innovation. All people should be able to participate in open societies and knowledge production. Therefore, it is essential to eliminate as many of these user psychological problems as possible.

With the rapid expansion of wiki use in organizations and between them, users are faced with problems in finding the core knowledge they seek. One of the most fascinating approaches in addressing this difficulty is the Semantic Web [45], which includes ontologies and reasoning on domains, as seen on the Gene Functions Wiki [15], Semantic MediaWiki, IkeWiki [19], and EKOSS [19]. As stated in [33], "We believe that by combining rich XML (DITA) structure, collaborative DocumentSpaces and wikis, we can help organizations break down the barriers that prevent them from achieving cross-departmental collaboration." This perspective can be expanded from the organizational context to the cooperation and co-work between organizations and to their whole supply chain [28], which is one of the sources of innovations made via cooperation.

Our continuing work involves an inquiry into the barriers of using these kinds of innovation portals the users encounter in their work. It is necessary to understand the various psychological hindrances people meet at the individual, group, organizational, and corporate levels in participating in decentralized innovation societies. In the future, such societies will be the most prominent sources of new, breakthrough innovations and the starting points for new economic growth in the global economy.

Acknowledgment

The authors express appreciation to the Finnish Forest Cluster for its favorable attitude toward our studies. We will also give thanks to the new, innovative company, Inno-w, the designer of this particular platform under investigation, for the possibility to use it as the research subject.

REFERENCES

[1] J. Anderson, John, M. Matiessa and C. Lebiere, "ACT-R: A theory of higher level cognition and its relation to visual attention." Human computer interaction, 12, 1997, pp. 439-462,

[2] M. Bernard, "Examining a Metric for Predicting the Accessibility of Information within Hypertext Structures." Dissertation Thesis, Wichita State University, 2002.

[3] S. K. Card, T. P. Moran and A. Newell, The Psychology of Human-Computer Interaction, Erlbaum, Hillsdale NJ, 1983

[4] B. Carlson and R. Stankiewicz, "On the Nature, Function, and Composition of Technological Systems", Journal of Evolutionary Economics, 1(2), 1991, pp 93-118.

[5] J. M. Carroll, "Human-computer interaction: Psychology as a Science of Design", Annual Review of Psychology, February, 48, 1997, pp. 61-83.

[6] Henry, W., Chesbrough, The New Imperative for creating and Profiting from Technology, Boston, Massachusetts, Harvard Business School press, 2003.

[7] P. Cohendent and F. Meyer-Krahmer, "Technology Policy in the Knowledge-Based Economy", in Innovation Policy in a Knowledge-Based Economy. Theory and Practice, L. Patrick and M. Mireille, Eds. Springer-Verlag, Berlin, Heidelberg, 2005, pp. 75-112.

[8] Ian Dey, Qualitative Data Analysis: A user-friendly guide. London: Routledge, 1993.

[9] M., Dodgson and J. Bessant, Effective Innovation Policy: A New Approach, London: International Thomson Business Press, 1996.

[10] EU's European Technology Platforms. [Online] Available: http://cordis.europa.eu/technology-platforms/home_en.html.

[11] The Finish Forest Cluster. [Online] Available: http://www.forestindustries.fi/infokortit/forest%20cluster%20ltd/Pages/default.aspx.

[12] J. Gwizdka and I. Spence, "Implicit measures of lostness and success in web navigation", Interacting with Computers, 19, 2007, pp. 357-369.

[13] M. G. Helander, Handbook of human-computer interaction, Eds. Martin G. Helander, Thomas K. Landauer and Prasad V. Prabhu, Amsterdam: Elsevier 1997.

[14] Sacha Helfenstein and Pertti Saariluoma, "Mental contents in transfer", Psychological Research, 70 (4), July, 2006.

[15] Robert Hoehndorf, Kay Prüfer, Michael Backhaus, Heinrich Herre, Janet Kelso, Frank Loebe and Johann Visagie, „A Proposal for a Gene Functions Wiki" R. Meersman, Z. Tari, P. Herrero et al. (Eds.): OTMWorkshops 2006, LNCS 4277, pp. 669–678, 2006. Springer-Verlag Berlin Heidelberg. [Online]. Available: http://www. springerlink.com/content/t535073n63336068/fulltext.pdf.

[16] Sanna Juutinen and Pertti Saariluoma, P. "Some emotional Obstacles of E-learning", Digital Learning India 23-25.8.2006, New Delhi, India.

[17] Sanna Juutinen and Pertti Saariluoma, P. "Usability and emotional obstacles in adopting e-learning - A case study", 2007 IRMA International conference, May 19-23, 2007, Vancouver, Canada

[18] J. Klobas, Wikis: Tools for information Work & Collaboration, J. Klobas, Ed. Oxford: Chandos Publishing, 2006.

[19] Steven Kraines , Weisen Guo , Brian Kemper and Yutaka Nakamura, "EKOSS: A Knowledge-User Centered Approach to Knowledge Sharing, Discovery, and Integration on the Semantic Web", Lecture Notes in Computer Science 4273/2006, Springer Berlin/Heidelberg, 2006. [Online]. Available: http://www.metapress.com/content/85617065848214g7/fulltext.pdf

[20] Markus Krötzsch, Sebastian Schaffert and Denny Vrandecic, "Reasoning in Semantic Wikis", Lecture Notes in Computer Science 4636/2007. Springer Berlin, Heidelberg, 2007. [Online]. Availabel: http://www.metapress.com/ content/g2jx102461w52456/

[21] Juha, Lamminen, Mauri, Leppänen, Risto, Heikkinen, Anna, Kämäräinen, and Elina, Jokisuu, "A quantitative method for localizing user interface problems", Submitted to Human technology, 2009. [Online] Available:http://www. humantechnology.jyu.fi

[22] Alan MacCormack, Theodore Forbath, Peter Brooks, and Patrick Kalaher, ‟Innovation through Global Collaboration: A New Source of Competitive Advantage", Working Knowledge, Harward Business School, 2007.

[23] James Stanley Metcalfe,. "Systems failure and the case of innovation" in Innovation Policy in a knowledge-Based Economy. Theory and Practise, Patrick Llerena and Matt Mireille (Eds.). Springen-Verlag, Berlin, Heidelberg, 2005.

[24] Raquel Navarro-Prieto and Jose J. Canas, "Are visual programming languages better? The role of imagery in program comprehension", International Journal of Human-Computer Studies, Volume 54, Issue 6, June 2001, pp. 799-829.

[25] U. Neisser, Cognitive Pyschology, New York: Appleton-Century-Crofts, 1967.

[26] L. Patrick and M. Mireille, Eds. Innovation Policy in a Knowledge-Based Economy. Theory and Practice, Springen-Verlag, Berlin, Heidelberg, 2005, pp. 1-11.

[27] T. Moran, "An applied psychology of the users," Computing Surveys, 13, 1, March, 1981, pp. 1-11.

[28] Oslo manual: Guidelines for collecting and interpreting innovation data. Measurement of scientific and technological activities, 3rd ed. Paris: Organisation for Economic Co-operation and Development Statistical Office of the European Communities, OECD, 2005. [Online]. Available: http://ulutek.uludag.edu.tr/downloads/Oslo_Manual_Third_Edition.pdf.

[29] A. Oulasvirta and P. Saariluoma, "Long-term working memory and interrupting messages in human-computer interaction," Behaviour & information Technology, 23, 1, Jan-Feb, 2004, pp. 53-64.

[30] A. Oulasvirta and P. Saariluoma, "Surviving task interruptions: Investigating implications of long tern mowrking memory," International journal of human computer studies, 64, 2006, pp. 941–961.

[31] S. Paquet, "Wikis in business," Wikis: Tools for information Work & Collaboration, J. Klobas, Ed. Oxford: Chandos Publishing, 2006.

[32] H. Parkkola, P. Saariluoma and E. Berki, "Action oriented classification of families' information and communication anctions: Exploring mothers' viewpoints," Journal of behaviour and information technology, 2006.

[33] Paul Prescod, "Blast Radius XMetaL, The convergence of structure and chaos." 2005. [Online]. Available: http://www.idealliance.org/ proceedings/xtech05/papers/03-02-04/.

[34] Teresa, Roberts, in Handbook of human computer interaction, M. Helander Ed. Amsterdam: North-Holland, 1988.

[35] P. Saariluoma, Käyttäjäpsykologia [User-psychology], Porvoo, Finland, WSOY, 2004.

[36] P. Saariluoma, (2005)."Explanatory frameworks for interaction design," A. Pirhonen, H. Isomäki, C. Roast & P. Saariluoma, Eds. Future interaction design, London, Springer.

[37] L. Salmeron, J. Canas, W. Kintsch and I. Fajardo, 2005. "Reading Strategies and Hypertext Comprehension" Discourse Processes, *40*(3), 2005, pp. 171–191.

[38] M. K. Singley and J. R. Anderson, The Transfer of Cognitive Skill, Cambridge, MA: Harward University Press, 1989.

[39] Strategic Research Agenda of the European Forest-based Sector Technology Platform, 2008. [Online] Available: http://www.forestplatform.org/.

[40] E. Styles, "The psychology of attention", Hove: Psychology press, 1997.

[41] Helena Suvinen, and Pertti Saariluoma, "User Psychological Problems in a Wiki-Based Knowledge Sharing Portal", Proceedings

of the Third International Conference on Internet and Web Applications and Services, (ICIW'08), June 8-13, 2008 – Athens, Greece. 2008, ISBN: 978-0-7695-3163-2, IEEE Computer Society, Washington, DC, USA, pp. 552-557.

[42] Don Tappscott and Anthony Williams, Wikinomics How Mass Collaboration Changes Everything, Penquin Group (USA) , New York, 2006.

[43] A. Treisman,. "Features and objects", Quarterly journal of experimental psychology, 40A, 1988, pp. 201-237.

[44] C. Wagner, "Wiki: A technology for conversational knowledge management and group collaboration", Communications of the Association for Information Systems, 13, 2004, pp. 265-289.

[45] Christian, Wagner, Karen, S., K., Cheung, and Rachael, K., F., Ip, "Building Semantic Webs for e-government with Wiki technology", *Electronic Government* 3(1), 2006, pp 36-55.

[46] C. Wagner and A. Majchrzak, "Enabling Customer-Centricity Using Wikis and the Wiki Way", Journal of Management information Systems, 23, 3, 2006,pp.17-43.

[47] C. Wagner and P. Prasarnphanich. "Innovating Collaborative Content Creation: The Role of Altruism and Wiki Technology", Proceedings of the 40th international Conference on System Sciences – 2007, pp 18-28. [Online] Available: http://csdl2.computer.org/comp/proceedings/hicss/2007/2755/00/27550018b.pdf.

[48] Graham Vickery and Sacha Wunsch-Vincent, Participative Web and User-Created Content. WEB 2.0, WIKIS AND SOCIAL NETWORKING. OECDpublishing, 2007. [Online] Available: http://213.253.134.43/oecd/pdfs/ browseit/9307031E.PDF.

[49] Wiki-based portal, Forest Cluster Research Portal, 2006, 11. [Online]. Available: http://www.forestclusterportal.fi/index.php/Main_Page.

APPENDIX 1

	Classification of the usability problems and proposals for solving them [41]
Concepts	1. Group facilitator and Support facilitator are not obvious concepts for the user.
	2. While making groups, the concept KR1 is not obvious for the user.
	3. There is no easy way of adding new information in correct form: The Help-section is needed to understand the task. There is confusion about the concepts of article, page, file and Wiki-type information. Only 4 of 14 subjects were able to contribute without the help of the researcher.
Content clarification	4. On the Search results page, the color of links and the color and size of fonts for the text are confusing. The primary heading must be clearer so users do not cast down their gaze.
	5. The main page of dictionary is necessary.
	6. The Category: Research group page is, at the moment, confusing. G, G Cont and T are not obvious for users.
	7. The purpose of the Special Pages page is not comprehensible to users.
	8. Technical pages or content generation and using of them are confusing.
	9. It is not obvious where the information is stored and how it is connected and linked.
Function clarification	10. The differences between the links on the left, at the top of the article, and at the top of the page are not obvious to users.
	11. There is no page map. The user should know his/her navigation point all the time.
	12. The difference between the function of the Go button and the Search button is not obvious to users. Experienced users use the Return key without thinking, thus it functions as the Go button.
	13. Scanning the Search Results page is confusing when there are not many pages. The results page should include information regarding the number of pages returned.
	14. The use of the namespaces at the bottom of the page is not comprehensible to users.
	15. The difference between Quick Guide and the Help page is not comprehensible to users
	16. On the Help page, the Creating New page steps 1, 2 and 3 are explained well, but are confusing, because the last step includes the text “This way isn’t recommended.”
	17. The picture on the User’s Guide page is illustrative, but it divides the page. The instructions mislead users to start adding groups and ideas, when they should add information.
	18. There should be clear hierarchies and routes to execute functions.
Page elucidation	19. The grouping of the information on the pages of the various communities is confusing. The information in the areas between lines should belong together.
	20. The headings on the Community Portal page are not clear: First a black heading, then the same in green.
	21. In the Help page before the Creating New page is the Editing Images & Files –section. Users do not scroll/read long texts.
	22. The site has too much text, and too little visualization that would make it easier to understand the functions available in this portal.
	23. The Welcome texts are not needed.
Function automation	24. The Research Group list should be updated automatically when new research groups are created.
	25. There should be an Add button for inserting new information that should link directly with a template where the users could write the content, and it should function in the WYSIWYG-mode. The application should at least ask about the text formatting before saving.
	26. There should be an online Help site with the option to jump directly to the sought place in the Help page.
	27. The cursor hovering on a link should give a Popup text with more information about the link.
	28. New links should be provided in blue and visited links in red, in line with the standard in Web interfaces.
Training	29. Many of the subjects would have stopped inserting information and tried to get help from someone else. There should be one trained expert for this application in every research unit.
	30. There is no working culture of this kind of wiki societies, so it should be created by training.
	31. The site contained so much text that only highly interested users would use it. A clear demo can make it easier to get a mental picture of it. There are several domains that can induce problems: Using Wiki, the FFCRP interface, and the domain of the forest sector. There is no active working culture yet for this kind of application. [However, this culture has evolved quickly since our experiment.)

Online Teaching and Learning – Developing and Using an eEducation Environment

Manuel Goetz, Stefan Jablonski, Michael Igler,
Stephanie Meerkamm
Chair for Applied Computer Science IV
University of Bayreuth
Bayreuth, Germany
(manuel.goetz, stefan.jablonski, michael.igler,
stephanie.meerkamm)@uni-bayreuth.de

Matthias Ehmann
Computer Science Education
University of Bayreuth
Bayreuth, Germany
matthias.ehmann@uni-bayreuth.de

***Abstract*— To support schools in teaching computer science, we have started the Informatik@School project. In this project, which is meanwhile funded by "Oberfrankenstiftung", we communicate computer science content to beginners and matured students. As the number of participating students is very large in comparison to the number of advisors and big distances have to be bridged, we separated students into two groups. Students from schools located not far from our university are taught in common face-to-face lessons, while far-off students get taught the same content in online lessons.**

In this paper we present the project, its preconditions and the didactical and content based concepts. We will introduce a web based technical environment which fulfills these issues and facilitates realization of afore mentioned online courses. Finally, we present lessons learned from this project and draw conclusions especially concerning the technical platform which consists of hardware and software used.

Keywords: *e-learning; online teaching; online education environment*

I. Introduction

Since the German curriculum at secondary schools was reformed, computer science is an independent subject. Teachers skilled in this discipline are rare as education of computer science teachers has just begun. Furthermore, computer science is a broad area and schools are limited with respect to time they can spend on computer science education. Consequently many interesting fields of computer science cannot be addressed.

Therefore a project named "Informatik@School" was set up (funded by "Oberfrankenstiftung") which supports schools in computer science education and should increase students' interest in computer sciences [1]. As this project is not limited by a curriculum, topics are freely selected; they should be of major interest for the students. Limitations of this project are the amount of time the advisors can spend and the distances between schools and the advisors. According to that some schools are taught remotely while others can be taught in common classes. In this paper we present our online teaching approach consisting of a technical and didactical part and compare success to traditional learning methods.

The rest of the paper is structured as followed: In Section II we give an overview about the didactical teaching and learning concept used in our approach. Section III provides a collection of requirements and their technical implementation needed for realization of the didactical concept. In Section IV we describe an application of our didactical and technical concept in computer science teaching and learning. Finally, Section V summarizes our experiences and provides and outlook to future improvements and research.

II. Teaching and Learning Concept

Our final goal is the development of an online environment for teaching and learning. To find an adequate solution it is necessary to analyze the situation of online teaching and learning and the development of a fundamental learning concept. After these steps are done, it will be possible to identify necessary parts of a software solution.

A. Teaching and learning situation

From the time of Johann Friedrich Herbart on, the didactic situation of teaching and learning can classically be described by the didactic triangle [23] (Figure 1).

It covers the interdependencies between teacher, learner and content. This figure visualizes the dependencies of the main factors of teaching and learning. Analyzing the didactic triangle can lead to different didactical concepts.

Brain research and psychology made large progress in understanding the human learning process and the underlying cerebral structures during the last century [30]. According to these results learning must be understood as an individual process. A learner builds his own cerebral web and embeds new knowledge into his web. The connections between “chunks” of knowledge are built during the learning process. The stronger these usage dependent connections are the better the fetch of the chunks works.

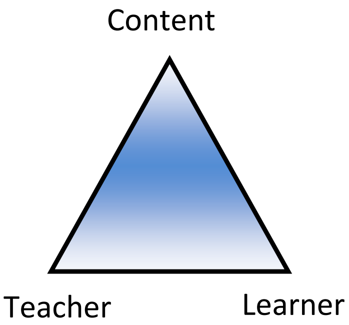


Figure 1. Didactic triangle

Taking all previous points into account we can conclude that learning is more successful if students play an active part in the learning process. The teacher is not the main person; he is in the role of an organizer of the learning environment. These results are similar to the constructivist learning theory.

The student centering is the most important part of our didactic concept.

B. Demands on the Future Generation

In didactics, demands on the future generations are often used to classify the content of teaching and learning. But demands on the future generation also influence the way of teaching and learning – the methodology.

Live long learning is one key word in our society. We must enable our students to learn self-dependently. They need didactical tools to make new fields of knowledge accessible to themselves. This is another motivation (see II.A) for a student centered learning concept.

Another key word is key skills qualifications / soft skills [26]. Employers and educators criticize that graduates are not well equipped with basic general skills which are necessary for their future professional life and full participation in society. Standardized assessments like TIMSS [17] or PISA [20] are tools to get an objective result concerning knowledge and key skills qualifications of students. They partially confirm the criticism.

The resulting demand on the future generation is the ability to solve problems. All other key skills qualifications are tesseras to succeed in problem solving. According to that we identify the main qualifications to develop a didactical model for sustainable learning and teaching:

Communication. Communication is the fundament in our work-sharing society. Students need to practice communication with others. They also need method competence in selecting and using communication tools.

Cooperating with others. Communication is just the key to cooperate with others. Students must learn concepts of cooperation like team play, constructive arguing and taking responsibility. They have to combine these skills with communication.

The concept of pair programming, for instance, from the extreme programming approach contains good ideas of cooperation in computer science tasks [3].

Presenting. Presenting work results becomes more and more important in all areas of work life. Students must acquire presentation techniques and get used to giving presentations in front of an audience.

Improving own learning and performance. As requirements will change much faster in future, we must equip our students with techniques of self dependent learning to give them the chance to adapt to any kind of changes. They must be able to identify targets and work towards them.

Acquiring these previously listed skills works best in social situations. They are all requirements for problem solving. Problem solving is and was an important qualification for any generation. Today the world-wide-web supports finding solutions for many issues. It is the most comprehensive knowledge base in history of mankind. But problem solving is more than investigating the web. Students need to gain an insight in problem solving strategies. A possible process for solving problems can be adapted from computer science. It consists of 4 steps: modeling, processing, interpreting and validating [9] (Figure 2).

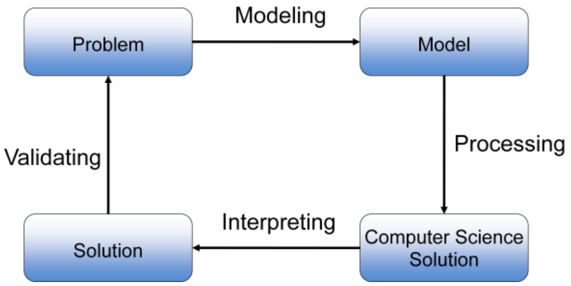


Figure 2. Problem solving

Modeling is necessary to make a problem processible:

- Analyzing the structure
- Identifying main objects, their properties and relationships
- Defining operations within a model

A model is a miniworld view of a given problem. Students need mechanisms like analyzing, structuring and abstracting to succeed in modeling.

Creating a model is just a first – but important – step. The model itself is not the solution of a problem –

for example – just as little as an entity relationship model is the final implementation of an enterprise resource planning system. Creating and working with the model helps to structure a problem and find a solution.

Processing the model leads to a first solution, which is produced based on the model created in the previous phase. In computer science, the processing phase can be an implementation in a programming language.

Interpreting the solution is a kind of inverting the modeling. It re-translates the solution from the miniworld's "language" back to the "language" of the entire problem. Considering an implementation, we interpret the result delivered and draw conclusions.

Validating the interpreted solution is the final step. It helps to identify errors in the whole process and is a kind of quality measurement. It is necessary to go through all steps to get a validated solution.

This process of problem solving has a recursive structure and is valid for problems from many disciplines. It can be applied to widespread problems; included smaller problems can be identified and solved using the same strategy.

Taking all results of our previous analysis into account, we have to implement a didactical concept for individual as well as cooperative problem-oriented teaching and learning in an electronic environment.

C. Structuring the learning process

We will introduce our teaching and learning concept by describing the structure of the learning process. Later on we will go into further details of important phases.

1) Teaching usage of communication tools

As education via online courses is uncommon to German students, first an introduction to the technical environment has to be given. Although some students were used to parts of our environment because of free time activities, no student was familiar with all tools of our online teaching environment (see III.D, III.F).

Independently of the tools that should be introduced, we could observe a huge cooperativeness of students to help each other in this phase.

2) Introduction to theoretical and practical concepts of teaching content

Basic concepts of the application domain are introduced. In case of computer science and web technologies, important concepts are the OOP (Object-Oriented Paradigm) [16] and especially the structuring and modularization of problems. Furthermore, an introduction to the applications that should be used for applying the learned theoretical concepts was given (Squeak [31] and Scratch [18] for beginners and Java with Eclipse WTP [7] or jMonkey Engine (JME, [25]) for advanced students). In this phase, consequent feedback from students is extremely important to achieve a good learning curve. Also direct and fast support is essential for motivating students.

3) Applying concepts learned in simple exercises

In order to train transfer and problem solving techniques, the introduced concepts are applied to simple problems. With supervised implementation of simple applications, students are forced to think about the concepts learned.

Although this is an extra phase, it cannot be completely separated from the previous phase as there is an overlapping; distinction is done concept and content based and not because of the timeline of teaching.

We found two factors which are especially important for success of this phase. Firstly, there is a need to avoid a higher degree of frustration at students. Therefore, consequent encouragement for asking questions concerning their (probably not working) solution has to be done. This has to be combined with fast and clear answers, when support is needed (see also [22]). Secondly, the possibility to "revisit" a lesson with watching a video-on-demand containing the lesson is used broadly.

4) Solving a daily problem single-handed

After applying the concepts learned on simple problems in a supervised environment, students work in teams to solve a daily and more complex problem. They should get a feeling for the complexity of common domain problems and a rough understanding for the different modules a structured solution is composed of. As a real problem of a domain can mostly not be solved from one lesson to another, about two months are given for finding a solution. Also the character of lessons changes. Advisors no longer have an active part, but answer questions on special problems. Furthermore, support should not be that detailed as in the previous phase, but only give a rough solution that then should be detailed by the students. This way a self dependent working style is trained.

D. A basic concept for individual and cooperative problem-oriented teaching and learning

Especially the realization of step 3 and 4 cannot be covered by traditional teaching. Here our approach for individual and cooperative problem-oriented teaching and learning is applied. A concept that fulfills the requirements is the "I-You-We" approach (Figure 3). It is a problem-oriented way of teaching and learning with three stages of individuality [33]. We adopted this concept to online learning and teaching (see IV.C, IV.D).

In the "**I phase**" the students are confronted with a problem and have to explore the situation by themselves. They should try to find an individual solution. In contrast to common teacher centered education, the students take an active part in the learning process: They go their own ways and they make their own decisions and experiences. The solutions may not be perfect, but also mistakes help them to improve understanding. For example, they are asked to create an object diagram to visualize the objects and relationships regarding the online chat. The advisor becomes some kind of coach or tutor.

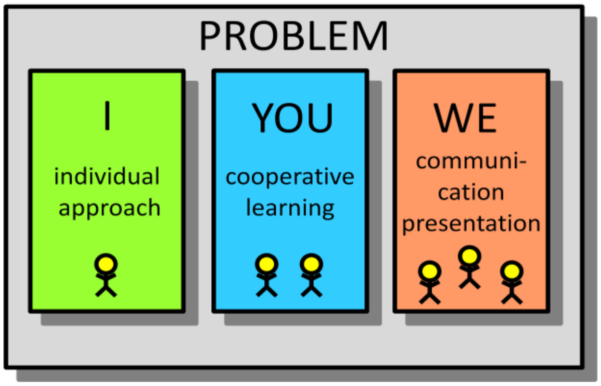


Figure 3. I-YOU-WE Concept

Cooperation characterizes the "**You phase**". Together with a partner, students rethink their solutions. Discussing about the problem and explaining their ideas support this process. They put thoughts into words to communicate with the partner. This is another active learning process. Errors can be identified and different solutions can be combined.

The collaboration with the partner is a kind of dress rehearsal for the presentation in the whole group and helps them to overcome their inhibitions.

The two phases of individual exploration culminate in the common presentation of the students' work, the "**We phase**". Some learners show their ideas to the whole group with reporting their results and difficulties. The listeners complete the remarks. As every student was engaged in the given problem in the two phases before, he should be familiar with it. The teacher is the moderator. He leads the discussion and he combines the gathered work with new content. Discussing and communicating about a situation deepens the understanding of the problem and opens each students mind for manifold approaches.

In our concept there are two possibilities for the last phase. The presentations can take place in on site lessons in every school. Then the local teacher moderates the session. The online alternative uses web technologies.

We apply this concept also to the on site lessons of the project. It can also be used in traditional teaching in different subjects as "islands" of self dependent and cooperative learning [6]. This can be the beginning of a new way of teaching and learning.

The duration of units covering the concept reaches from parts of a single lesson up to projects lasting several weeks.

III. TECHNICAL ENVIRONMENT

After introducing the teaching and learning concept, our concrete realization is described in detail.

A. *Preconditions*

Goal of our project is to support schools in computer science education [10]. Participation of students is optional and they can leave the project every time. The curriculum should be interesting and directed to students' interests in order to increase their motivation to deal with and apply computer science techniques. This means we can communicate content concerning computer science in a form students appreciate.

In this project more than 200 students per year from 15 schools participate and are attended by only three advisors. Three of these schools with about 30 - 40 students can be visited directly; students of the other schools need to be taught remotely because of the local distance which is up to 70 kilometers. For online teaching we use an internet based e-learning approach [4].

In our online schools local mentoring is needed, especially for setting up and introducing students to the online environment. Therefore a teacher of every participating school is prepared for usage of technical environment and is taught basics of the lessons' content.

B. *Requirements to an online teaching environment*

There are manifold requirements for an online teaching environment (OTE). During the first two years of our project Informatik@School we gained much experience in the field of e-teaching and gathered information for a requirements review. We can categorize the demands in three scopes: functional requirements from students, functional requirements from advisors and technical requirements. The functional requirements take into account the demands of our learning concept (Section II).

Functional requirements from students. From students' point of view, an OTE has to be a tool easy to use. In our project students have the chance to participate in online lessons in all places. Consequently an OTE should be available at school and at home. It has to combine several single applications in just one user interface and should cover several use cases.

We offer live online lessons. During these lessons students should see advisor's desktop and hear his voice. Students should also have the possibility to ask questions during a lesson (see II.B). This requires unidirectional video and bidirectional audio transmission. To arrange an almost face-to-face learning situation for our students it would be nice to have a web cam transmission of the advisor.

During online lessons, students work with their own project files. These files are normally stored on their local PC, but in our use case more flexibility is needed as students may start a project at school and finish it at

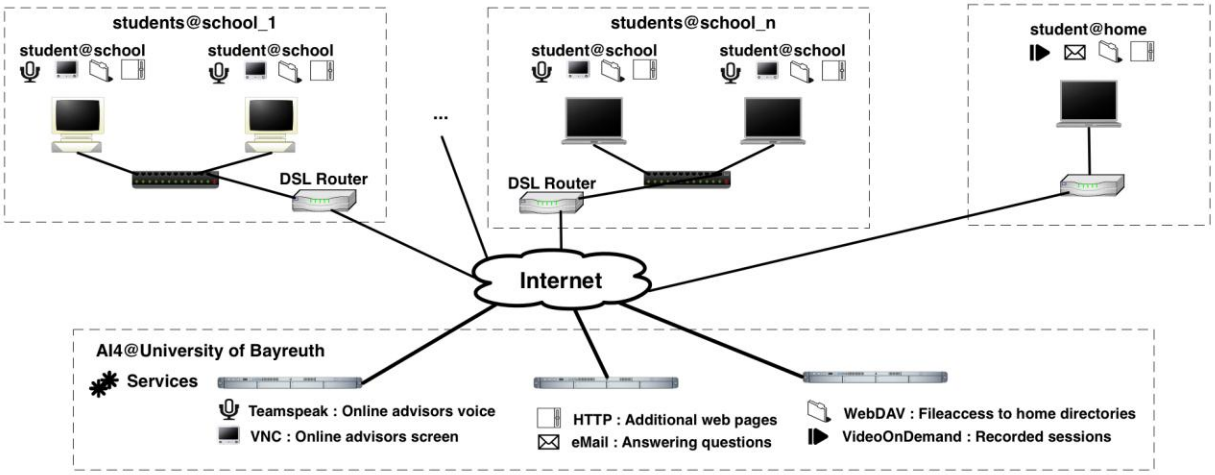

students@school_1
student@school
student@school
...
students@school_n
student@school
student@school
student@home
DSL Router
DSL Router
Internet
AI4@University of Bayreuth
Services
Teamspeak : Online advisors voice
VNC : Online advisors screen
HTTP : Additional web pages
eMail : Answering questions
WebDAV : Fileaccess to home directories
VideoOnDemand : Recorded sessions

Figure 4. Overview of first topology

home. To enable seamless portability of files, an OTE should offer an online storage solution for students' files called data pool. This data pool has to be accessible from anywhere and should be independent from the computer students currently use. The online storage solution is also important for collaboration of students (see II.B). We want to encourage our students to exchange their ideas and work together in groups (see II.B, II.D). So they need the possibility to provide access to their files for other students. An OTE should offer a simple right management solution for the data pool.

The cooperation between students can be real or virtual. For virtual cooperation we need forums and online chat functionality. Some students do not have the chance to participate in the live online lessons; other students want to revisit a lesson. For these participants we should offer video streams of all online lessons.

Functional requirements from advisors. The requirements of communication during and after online lessons are also valid from advisor's view. The live audio communication should be under advisor´s control. He manages the voice rights and grants voice privileges to students after a request. This is a way to offer the possibility of asking questions and giving a lesson in a controlled way.

Many questions of students can be answered easily without watching what students have done at their PC. But especially in the phases in which student's work on larger projects (see II.C.3, II.C.4), questions and problems become more and more complex. In these situations qualitatively good and fast support can only be given if the advisor can see and also access students' desktop. The advisor also needs a storage solution for his files. He has to publish project files and scripts for all students.

An OTE has to provide a tool where students' exercise solutions can be collected. It should be a central repository where the advisor can access the files, correct them and provide some comments for the students.

Technical requirements. Beside the functional requirements of the users we also have to keep the technical requirements in mind. Especially the available hard- and software equipment at schools can be a limiting factor.

The client environment is very heterogeneous. Students use different PCs with different operating systems at school and at home. We can meet with this obstacle by using portable client software on students' side.

Also security restrictions at school have to be faced. The client software has to run with standard permissions. Network applications have to use standard ports.

Taking all this into account, a web application is the best solution to fulfill the technical requirements.

Furthermore the client application needs a user-friendly installer to support PC administrators at schools in the best possible way and facilitate installation for students at their PCs at home.

C. Technical Environment at schools

Main limitation in designing the technical environment was a heterogeneous technical infrastructure at the participating schools. We had to consider very different hardware configurations ranging from up to date computers to PCs that are aged more than 10 years. The same applies to the used screens, i.e. we needed to be compatible to quite low resolutions on students' side. Additional to the hardware, already installed software on the computers had to be regarded. On the one hand, schools work with different operating systems (and some schools even use very special configurations of an operation system). On the other hand, compatibility of our software to installed security software needs to be provided. Especially schools have very hard restrictions concerning security which leads to school environments where students cannot save files

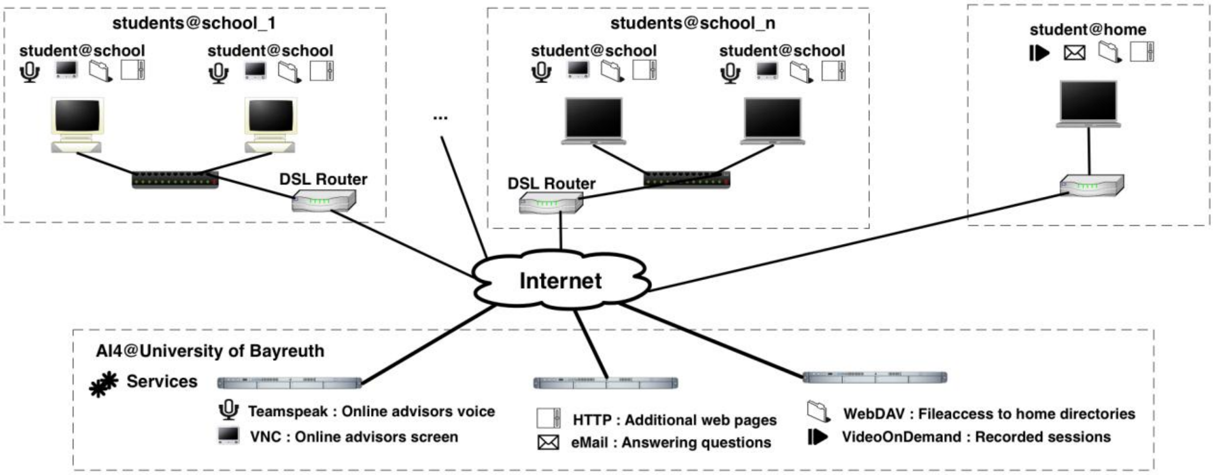

students@school_1
student@school
student@school
DSL Router
...
students@school_n
student@school
student@school
DSL Router
student@home
Internet
AI4@University of Bayreuth
Services
Teamspeak : Online advisors voice
VNC : Online advisors screen
HTTP : Additional web pages
eMail : Answering questions
WebDAV : Fileaccess to home directories
VideoOnDemand : Recorded sessions

Figure 5. Overview of the eEE topology

or just have very restricted access to the internet (which we need to use to provide communication). Consequently, we had to realize a platform spanning solution on client side.

As we are in the second run of our Informatik@school project meanwhile, we have a first implementation of this software (Figure 4) which was used in the first term and an improved second version which we are using currently (Figure 5). Both versions will be presented in the following and the lessons learned during usage of our first version will be discussed as these lessons lead to the implementation of our second version.

D. *Online Teaching Environment Version 1.0*

To provide an audio communication channel during online sessions, we decided to install a VOIP (Voice Over IP) environment. After evaluating different systems, our decision fell to the free application (for non commercial entities) “Teamspeak” [32]. This solution is available on common operating systems like Microsoft Windows, Apple OS X and Linux. It provides a spanned solution for the most configurations on the client side. Due to the web based administration control panel and the highly scalable user permissions system, it is very flexible and comfortable to administrate the accounts in our server environment. During the online course all students hear the advisors voice and we can grant a student the right to talk if he is requesting voice for a question. All other participants in the online course can hear his question.

Screen content of the advisor's computer is transmitted via a VNC-server [27] to the VNC-clients on the classroom side. Students can watch the transmission live on their own screens or on a video projection in the classroom. Accessing the transmission is possible via the VNC server's integrated web service. So a connection using a java-capable browser is possible, which avoids installing a VNC client on classroom computers.

Online lessons are recorded containing advisor's and students' voice as well as screen content of the advisor. It is saved in the windows media format WMF [36] which are provided as on-demand video streams. Video streams are accessible on our website and students can use them for postprocessing the online lessons at home.

Resources of the online lessons like PDF [1] files of presentations or programming libraries are downloadable through our website. So they are readable and presentable for any later references.

As data exchange tool between students and the advisors at university, we use a WebDAV [35] system in this version. WebDAV is an extension of the Hypertext Transfer Protocol (http). It allows bidirectional file transfer. The WebDAV service is accessible with login and password through the Internet Explorer [13] by entering the URL [34] of our WebDAV server. WebDAV folders are mapped to the file explorer of windows automatically. So this procedure presents the folder structure in a well-known way.

All necessary files of each lesson (WMF, PDF, project resources) are also stored on the WebDAV system; so the online lessons can be reused later for reference. Consequently the WebDAV system is a central storage unit for all course resources.

Questions occurring after a lesson can be asked by email or phone. Especially questions addressing students' problems with their current implementation are mostly asked by email as students can send their current source code in the attachments or as a link to their WebDAV folder.

E. *Lessons learned*

Almost every school network has a high bandwidth asynchronous DSL [29] connection to the internet, which is mostly secured by a firewall [21]. Although there is no standardized bandwidth of schools' internet connection, the minimal configuration of approximately 1 MBit/s downstream was sufficient to receive voice and screen content transmission during our online sessions without latencies.

The upstream bandwidth of 128kBit/s with asynchronous DSL connections was adequate to receive students' questions during online lessons in good quality. Additionally, it is desirable to switch to the computer of a certain student in order to help him solving a problem with his development environment or source code during the online lesson. Our didactical concept could be realized better if this kind of feedback would be available, too.

In the beginning of the project most common problems in initiating connections between schools and our server came up from different policy restrictions of the firewalls installed at schools. As long as there is a stateful firewall [21] it is not necessary to open any incoming ports. For more restrictive firewalls some configuration effort has to be done to enable a problem-free transmission.

Students had no basic problems concerning the usability of our environment. Switching between the windows of several stand alone applications (Web Browser Window for WebDAV, Web Browser Window for VNC video transmission, Teamspeak) was a bit unpleasant sometimes.

Based on our experience during the first run of the project "Informatik@School", we designed and realized an all-embracing solution. This new environment helps to minimize client side software installation effort by using web services. Furthermore it offers a single user interface for all services to avoid switching between several applications.

F. eEE – The next step in online teaching

Finally, the developed system for online teaching called eEE (eEducation Environment) is presented and discussed. In comparison to the version presented in [1] and III.D, progress regarding of some important features could be done. Firstly the functionality offered to the user could be integrated into one single web application which facilitates and accelerates the access to the available functionalities and makes learning and teaching more comfortable. Furthermore we succeeded in implementing the access to student's desktop by the advisor, which improves the quality of teaching a lot.

1) Conceptual layout of eEE

We decided to deal with the requirements mentioned in Section III.B by creating a web application. This application is deployed on a dedicated server which is connected to the internet and provides the server components of a client-server architecture. Students and advisors log in to this environment by login name and password through a web browser. eEE can be accessed from all schools and from students' homes.

Figure 5 gives an overview of the technical aspect of the architecture of our eEE. Students' computers are placed in the top part of this diagram. Students can be connected to the internet directly or through a local area network using a router. When a connection to our server is established and users are identified, students can use all services provided by our server software like audio conference, desktop transmission or functions independent of online lessons like chat or forums. In comparison to the first version (see 3.5.) now all services are integrated in one single environment. Regarding the additional functionality the advisor needs, it is valid as well. He just needs to connect to the server to give a lesson or use other features the software is providing.

eEE itself is now based on the tool Moodle [19]. Moodle is a learning management system focused on course management, but lacks most features needed for live online teaching. It provides a widespread plugin structure, i.e. some components can be added or exchanged easily without having any effect on the rest of the system. We developed a new plugin that supports online lessons to adapt Moodle to our needs. As we used this plugin structure we established a loose coupling. Updates or other extensions do not interfere with our plugin and vice versa, i.e. we are not restricted to one special version of Moodle.

With Moodle and the additional plugin for the online lesson we have an integrated concept for online teaching and a direct and facilitated access to all the functionality is possible.

2) eEE for students

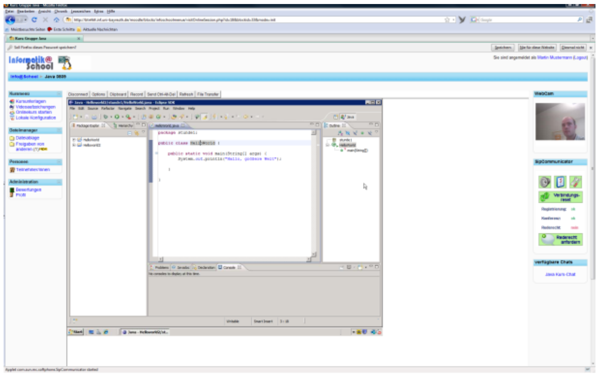
As already mentioned, in this version of our online teaching system the entire functionality is offered in one single system. Other systems are not necessary anymore.

At first, students need to select a course to login. After authentication, students can select many different options and have access to all working materials like scripts etc.

Now they can download these resources or open them in their browser. Additionally communication tools like a forum or a chat are available. This all time available communication platform raises collaboration between students and establishes a community in our project, although students may be separated by potentially big distances.

Students often need to switch their PC and working place (they may work at home, at school or meeting at a team member). Having the right data at the right moment on the right place was not easy to organize and handicapped the work a lot. Thus in this version we also offer an online storage which is accessible with our application. Students can upload and download files in their personal folder. They can also share files with other students; this sharing is based on the accounts, i.e. for every file or folder that should be shared the privileges to access this file or folder can be given to other students individually. That way teamwork in small groups is facilitated a lot.

The features presented so far deal with the administrative part of our environment, i.e. with the distribution of data and students' communication between the lessons. Lessons are available in two modes: students can participate in currently given online lessons or they can watch videos of former lessons. All lessons are recorded with sound and desktop of the advisor. These videos are available as streams and can be watched directly in the students' browsers. In older browser versions additional plugins need to be installed to play these video streams. Current versions of Internet Explorer or Mozilla Firefox, for instance, can start the videos directly as necessary video plugins are already integrated. To support a broad variety of systems, also links are given to access each stream from any potential player.



```
package stunde1;

public class HelloWorld {

    public static void main(String[] args) {
        System.out.println("Hallo, grosse Welt");
    }
}
```

Figure 6. Screen of students during online lessons

While an online course is given by an advisor, students can participate without prerequisite software installations (see Figure 6). In order to minimize requirements and to guarantee portability between systems, all components needed for online lessons on client side are based on Java Applets [12]. First, students can watch the advisors desktop. This is done via VNC streaming [27] and a Java client on students' side which is started automatically as an applet when students participate in an online lesson. Using the same approach, students can see the advisor as a webcam stream is delivered to the students. Lastly, students can listen to the advisor. This is managed using an online conference tool called Java Voice Bridge [14] which was developed by Sun. We integrated this tool and implemented a new user interface. Joining an online course, a student also joins the corresponding audio conference. This happens automatically so that students do not realize it. By default, students have no voice privileges, i.e. they can listen, but their voice is not transmitted to the other listeners. Therefore, students also have a button to ask for voice privileges. When this permission is granted by the advisor, students can ask questions; all other participants can listen to that question and the advisor's answer.

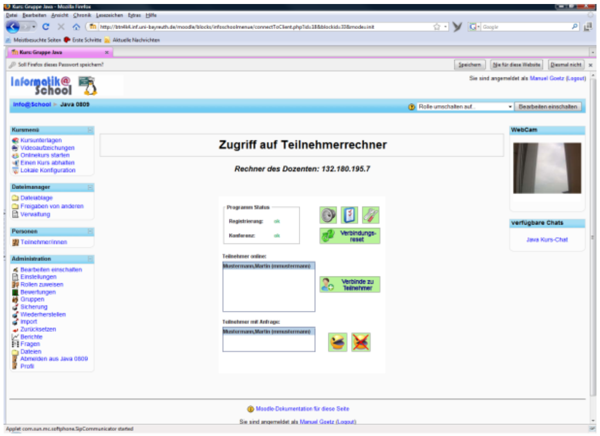
As we teach computer science, software on students' side is needed (currently JDK [8], Eclipse [11] and Scratch [18]). To provide this software, we offer an installer created with InnoSetup [28]. This installer also contains a VNC server which is needed for the optional access of an advisor to a student's desktop.

3) *eEE for advisors*

After discussion of the students' options in our system, the features for the advisors are presented. As for the students, in this version all features can be offered on one single platform which makes teaching much more comfortable. Furthermore we succeeded in implementing the access to students' desktops for the advisor during an online lesson. This was one of the requirements regarded as very helpful.

First, advisors can create courses. A course complies with a subject in school in our understanding. Courses may have one or more advisors who have the privileges to add scripts or new videos of lessons. Furthermore they can initiate online lessons.

After an advisor started an online lesson, his desktop is transmitted to the students automatically. This happens with a VNC client that streams the desktop to all registered clients. If a student participates in an online lesson, this registration is done automatically. The advisor can observe which students are currently visiting the lesson (see Figure 7).



http://btn4k4.inf.uni-bayreuth.de/moodle/blocks/infoschoolmenue/connectToClient.php?id=18&blockid=33&mode=init

Soll Firefox dieses Passwort speichern?

Sie sind angemeldet als Manuel Goetz (Logout)

Zugriff auf Teilnehmerrechner

Rechner des Dozenten: 132.180.195.7

Programm Status

Registrierung: ok

Konferenz: ok

Teilnehmer online:

Mustermann,Martin (mmustermann)

Teilnehmer mit Anfrage:

Mustermann,Martin (mmustermann)

Sie sind angemeldet als Manuel Goetz (Logout)

Applet com.sun.mc.softphone.SipCommunicator started

Figure 7. Screen of advisor during lesson

Via the same interface, the advisor is informed when students ask for voice privileges. Then he is able to grant this privilege to one or more students. He can also revoke this privilege individually after answering the question.

A feature that is extremely helpful, but technically hard to implement is the access to the students' desktop for the advisor during an online lesson. It is helpful as problems students currently have can be identified more effective if the advisor can really have a look on students' manuals. Consequently frustration for students is diminished. Usually access to students' desktops is forbidden by security restrictions. Furthermore, most students' PCs are placed behind some firewalls and/or routers and they do not have an own public IP address that could be addressed. Therefore, there is only one solution to provide this feature: the connection has to be established from the student's side, but without any action that needs to be done by the student himself. The advisor is the initiator.

If an advisor wants to access a student's desktop, a flag in a database is set. The database containing that flag is polled periodically from all participating clients. If this flag is set, then the student's browser starts a VNC streaming server and streams the students desktop to the advisor. As the connection is established from inside the student's network, there are no problems with firewalls. When the connection is not needed anymore, the flag is reset in the database and the connection is closed from the student's side. So there are no security gaps remaining.

4) eEE User interface

Finally we present the eEE user interface integrating the entire functionality. Layout for students, advisors and administrators is quite similar to facilitate usage with a clearly defined structure; the workplace is divided into three columns (see Figure 8):

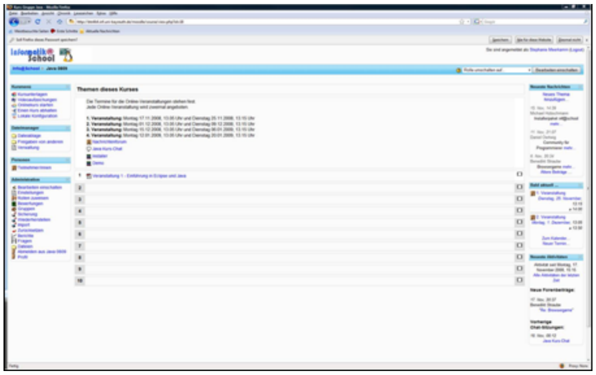


Figure 8. Screen of the workplace

- The left column offers a list of different activities and navigation possibilities. It is classified into "Course Menu", "Data Manager", "Participants" and "Administration". With the "Course Menu" the functions for the online-teaching are offered, as for example the courseware, the online session or the video recording. The "Data Manager" handles uploads and downloads of data files together with sharing of files between users. An overview about who is attending the course is given by the entry "Participants", whereas "Administration" offers general configuration possibilities.
- The middle column displays the content of each navigation topic. This can be all currently available scripts or, during an online lesson, the advisor's desktop.
- The right column contains additional information like news or next appointments. During an online lesson, the web cam shows the advisor.

This layout structure is always the same, whereas the offered activities in the left column change due to the role a person is logged in with. For instance, an advisor has the possibility to start an online lesson whereas students have the possibility to participate in a running lesson.

IV. Realizing the Concept with Computer Science Content

A. General organization of our project

We offered two courses for students aged between 14 and 19. While younger students participated in a Squeak eToys [31] or Scratch [18] project, the older ones took part in a Java (e.g. [8]) project using the Eclipse developing framework [11]. In every group we had approximately 100 students at the beginning.

Due to introduction of a new curriculum, students participating in the Squeak / Scratch group already had some basic knowledge concerning OOP (Object-Oriented Paradigm) [16] and control flow modeling. The members of the Java group had no preliminary knowledge in computer science. In the face of the different precognition levels, the teaching concept described above was applied to both courses.

We offered one session for every course in two weeks at local schools which could be supervised directly. For online schools, an hour every week with one repetition was offered, i.e. every second week was a repetition of the preceding week. As we had different time slots at repetitions, more online schools could participate.

B. Preparation of teachers

Although most courses are held online there is at least one teacher in every participating school for supporting the students on site. On the one hand this assistance is necessary because of the differences in software installations, network environments and rights management. On the other hand technical problems concerning the infrastructure in schools during an online lesson can be solved much easier on site which is an important factor for success of an online teaching project ([37]). In order to enable teachers to deal with these problems, they were prepared in a one day face-to-face course at university.

First the course concept and the technical preconditions were introduced to them. In the second part teachers got to know the technical environment and the installation of the necessary software products. The final and most extensive section of the in-service teacher training addressed the computer science content of the project.

Brief presentations gave a first overview about the aims. The content of our lessons was introduced and teachers dealt with the exercises of the first third of the online courses.

C. *Courses with Squeak / Scratch*

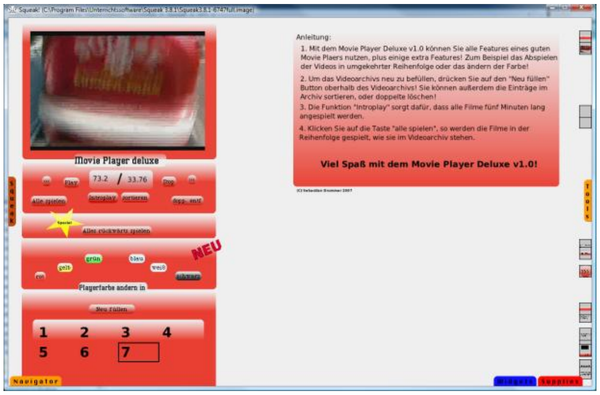
The curriculum for our younger students emphasizes basic, long lasting computer science concepts. As these students already have basic knowledge in Object Oriented Modeling (OOM) we use this as a connection point. We want to improve their knowledge in OOM and introduce object oriented and general programming concepts. To keep motivation high we have to combine the concepts with an attractive topic. The last two years we have chosen the field of computer games. All students have experience with computer games and we offer the possibility to look behind the scenes.

We used Squeak eToys during the first year and Scratch in the second year. Both development environments are based on Smalltalk [15] and offer an aged-based user interface for (object oriented) programming.

We apply our concept presented in 2.3. and 2.4. to our courses. During the first lessons we introduce the technical environment to the students together with basic computer science content. This makes the introduction of the tools more target-oriented. For instance, students have to download a draft of an animation film project from the central file repository; they “program” their own animation film and upload it as an exercise. The advisor provides a comment in the eEE and students can retrieve this comment. During the online lessons students use audio communication and chat to interact with the advisor. We observed no problem of the students concerning usage of eEE.

Also during introduction of new computer science concepts we try to make the online lessons student centered. They are always called upon to try out new concepts. They have time to work on their own and asked to give feedback. Always students have to present or describe their solutions. We also use exercises where students have to cooperate. For example, they have to model the concept of a car racing simulation and write a to-do list.

We inure students to be in a more active role during lessons. So the step to the third stage of our concept (II.C.3, II.D) is not that big. In this phase tasks become more complex and students have to apply and acquire knowledge. One of the tasks is to develop their own version of the arcade game “Pong” [24]. We offer a version they can use for playing but they cannot look behind the scenes. Each student has to develop a model of the game (I phase). He discusses and improves the model together with a partner (You phase). The “pair” implements the model together. Some of this work is done during online lessons. Students can ask the advisor for help if they have problems. They continue at home and ask for feedback using eEE tools. Finally the groups present their results (We phase). The presentation can be offline in their class or online with all students. The problems students solve in this phase help them to acquire new computer science concepts related to their present knowledge, i.e. they discover different kinds of condition-controlled loops and learn to use them. The kind of working according to II.C.3, helps students to improve their key skills qualifications.



Anleitung:

1. Mit dem Movie Player Deluxe v1.0 können Sie alle Features eines guten Movie Plaers nutzen, plus einige extra Features! Zum Beispiel das Abspielen der Videos in umgekehrter Reihenfolge oder das ändern der Farbe!

2. Um das Videoarchivs neu zu befüllen, drücken Sie auf den "Neu füllen" Button oberhalb des Videoarchivs! Sie können außerdem die Einträge im Archiv sortieren, oder doppelte löschen!

3. Die Funktion "Introplay" sorgt dafür, dass alle Filme fünf Minuten lang angespielt werden.

4. Klicken Sie auf die Taste "alle spielen", so werden die Filme in der Reihenfolge gespielt, wie sie im Videoarchiv stehen.

Viel Spaß mit dem Movie Player Deluxe v1.0!

Figure 9. Student's solution – Video player

In the fourth phase (II.C.4) students work in a project for about two months. One of the problems they have to solve is the development of a video player in Squeak (Figure 9). The task is constructed in a way that teamwork is necessary. Students organized themselves in teams with two or three members. They were responsible for the whole project management. Most of the work is done at home. We offer online lessons in this phase as well. Students have the chance to ask questions. The access to students’ PCs is very helpful for the advisor so he can identify problems much faster and provide more precise help. In the last runs of our project, students' solutions partially exceeded requirements. We recognized that students really cooperated and applied previous learned computer science skills as well as soft skills.

D. *Courses with Java*

For students in higher grades the Java course is offered. Goal of this course is to give an introduction into computer science contents through usage of professional tools and concepts.

In the first phase (see II.C.1), students need to get to know the communication tools which in our case is eEE (III.F). There are 3 sections that are especially interesting for students at the beginning:

- Resources that contain teaching material can be downloaded

- Online lessons can be participated through the system
- Communication functionality to the advisors and other students is provided by chat, forum, private messages etc.

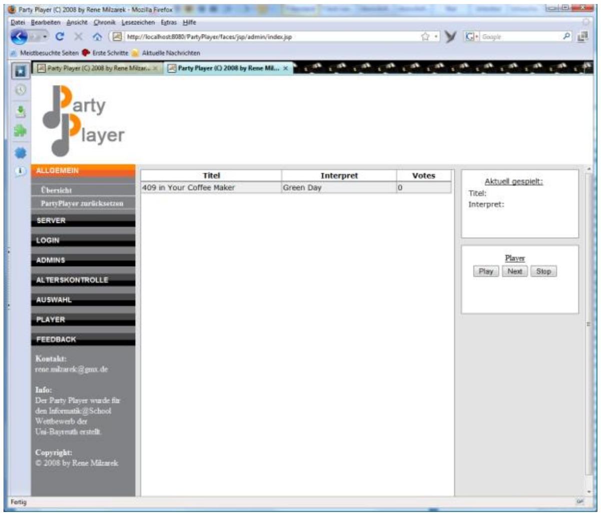
Students get introduced to the environment by their teachers after they visited the preparation course. This introduction is mostly quite short in time as many students are used to work with some kind of electronic communication tools. Therefore, they succeed by mainly following their intuition.

In the second phase, students are taught theoretical and practical concepts of the course content. In this course, they learn basic OOP concepts like information hiding, modularization or generalization / specialization. These theoretical concepts are directly used in the 3rd generation programming language "Java" in order to visualize them and show the effects. Java code is created in the professional IDE Eclipse [11]. Of course, a basic introduction to such a powerful tool also must be given to the students. All this content is also taught remotely in the online courses, i.e. we are just using our environment eEE to communicate to the students as advisors.

In the third phase, students practice concepts they have already learned during the second phase in order to raise the learning effect. Therefore, small examples are programmed which include a “HelloWorld” program or some kind of Banking Account example. In this example, for instance, students need to program an account at a bank as a class which fulfills several requirements. It needs to provide data fields for the account number or the account balance, which should be hidden in the implementation (information hiding). Furthermore, different kinds of sub-accounts should be implemented for families or companies (generalization / specialization). While implementing these small examples, students mainly reflect actively what they learned before and this deepens their understanding.

In the last phase, students should use all the contents of the course so far to manage one big problem. Typically, they have 8 to 10 weeks to provide a solution; teamwork is favored a lot in this phase. As an example, we wanted the students to implement a multi-user MP3 player which could be controlled through many workstations. Before, we taught them the usage of the web framework JSF [5] with the programming tool Eclipse WTP [7]. Figure 10 presents a screenshot of one of our students' solution. Different paths to the folders containing music can be configured. Furthermore, different strategies can be chosen to merge different playlists from different users. It is possible to merge them having equal rights or setting priorities as well as to configure the number of songs that needs to be waited until a song can be repeated.

In the end, students had to present their solutions in order to get feedback from other students or (through watching other solutions) getting more insights what could have been solved more elegantly in their own solution.


Party Player (C) 2008 by Rene Milzarek - Mozilla Firefox
Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe
http://localhost:8080/PartyPlayer/faces/jsp/admin/index.jsp
Google
Meistbesuchte Seiten Erste Schritte Aktuelle Nachrichten
Party Player (C) 2008 by Rene Milzar...
Party Player (C) 2008 by Rene Mil...
Party Player
ALLGEMEIN
Übersicht
PartyPlayer zurücksetzen
SERVER
LOGIN
ADMINS
ALTERSKONTROLLE
AUSWAHL
PLAYER
FEEDBACK
Aktuell gespielt:
Player
Play
Next
Stop
Fertig

Titel	Interpret	Votes
409 in Your Coffee Maker	Green Day	0

Titel:
Interpret:

Kontakt:
rene.milzarek@gmx.de

Info:
Der Party Player wurde für den Informatik@School Wettbewerb der Uni-Bayreuth erstellt.

Copyright:
© 2008 by Rene Milzarek

Figure 10. Multi-user MP3 Player

Altogether, we observed a very high level of the provided solutions. Furthermore, the students' ability to structure and solve problems was obviously very considerable and they only needed very little help to build up own solutions.

V. EXPERIENCE OF REMOTE TEACHING

A. Technical Environment

Main goal of a technical environment used in a school project must be usability. We increased usability a lot by designing and developing eEE. Using this software, most obstacles for students from a technical point of view were eliminated. Additionally, it enables us to realize the didactical concept in a more appropriate way.

B. Preparation and Organization of Online Teaching

The in-service teacher training turned out to be an excellent preparation of the teachers who care for the students in the “online schools”. They are integrated in the project from the beginning. Most of them take part in the online lessons and some also offer additional mentoring for the students.

Together with the teachers we organize information meetings in every school. During these meetings we introduce our courses to all interested students. This face-to-face event is very important for the students to meet their online advisors on site.

The students enrolled for the courses using an online form on our website. We collected all registration data in a database; this data was used to create accounts automatically.

It is not possible to bring all online students together on one fixed day. We offer each online lesson twice on two different weekdays. The video streams of recorded lessons can be used by students who cannot take part in online sessions; these streams are also used frequently to revisit lessons. Additionally students can ask questions by email, phone, chat or forum.

After the first third of the course we visit all online schools to intensify the personal contact with the students. This helps to hold up motivation and it is a useful feedback for us as online advisors.

C. Assessment of Online Teaching

As mentioned above up to three schools take part in the project in on site lessons. These groups play an important role in the online concept. Before an online lesson is held we use the corresponding on site lesson as a kind of dress rehearsal. The advisor receives direct response from the students and recognizes problems in the learning process. These are important experiences for the online lesson.

To nearly all students our student centered approach described in the “I-You-We” was completely new. They mostly had no experience in solving more complex problems on their own or in cooperating with a partner during a lesson. But after some lessons we observed in the on site classes that students started working together. Although each student had his own computer they sat in front of one computer in groups of two or three students. They started to discuss about the exercises and began to cooperate. After planning a strategy for a solution all students of a group returned to their own computer and started their work on a sub-problem.

In online classes this process took much longer. In the beginning it was harder for them to cooperate in small groups during online lessons. But the less information they received from the online teacher the more self-dependent they became. In the end of the third step “Applying concepts learned in simple exercises”, described in II.C.3), there was nearly no difference between the two groups. Some online students described their way of cooperation during a meeting at their school. They explained that they also met in their spare time to solve the exercises in groups.

The on site classes can also be compared with the online classes directly. At the moment this is more an informal comparison than a statistical assessment. The number of participants is too small to receive valid results. A statistical evaluation is already planned for the next year project.

But also the subjective experiences show interesting details.

The solutions of exercises show no significant differences in quality. Although we expected a discrepancy in speed of learning between the two groups we could not confirm that. The withdrawal rates for the volunteer courses do not differ.

“Online students” had no problems using our learning environment. Some students seem to have stoppages asking questions during the lessons using the voice communication tool. They mostly use the chat function.

VI. Conclusion and future Work

The technical concept of our e-learning environment follows the content and didactical concept (“I-You-We”). Each component of our infrastructure plays a dedicated role in teaching and learning. We use server based web services to minimize client side problems. Based on our experiences, we integrated bidirectional real time audio, web cam and screen content transmission and file access in one web based e-learning environment as a Moodle [19] extension.

Our experience shows that student centered learning is successful in online teaching. To achieve this we had to lead the students from a more teacher centered to a self dependent way of working step by step. Our technical environment builds the base for communication and cooperation of students among each other and with the online advisor.

Nevertheless, face-to-face contact between advisors and students should not be underestimated. It is necessary for students to hold up motivation and to get personal feedback. To lessen this disadvantage we implemented web cam transmission of the advisors themselves during online lessons to personalize the contact between students and advisors.

Also an on site contact person is very helpful to solve problems concerning local conditions. These local teachers are involved in the whole project, accompany the students and give feedback to the online advisors.

In future, we plan to adapt our teaching and learning concept in combination with our technical environment eEE to other domains; for instance, university language courses or further education at operational level in companies can be realized with our approach.

References

[1] M. Goetz, M. Ehmann, S. Jablonski, and M. Igler, "Experiences in Online Teaching and Learning", First International Workshop on Virtual Environments and Web Applications for e-Learning, with ICIW 2008, IARIA, Athens, 2008.

[2] Adobe PDF, http://createpdf.adobe.com, last revisited 2009-05-11.

[3] K. Beck, Extreme Programming Explained: Embrace Change, Addison-Wesley Longman, Amsterdam, 1999.

[4] Z. Berge, "Computer - mediated communication and the online classroom in distance learning", Computer-Mediated Communication Magazine, Vol. 2, Number 4, Hampton Press, Cresskill (NJ), 1995.

[5] H. Bergsten, JavaServer Faces, O'Reilly Media, 2004.

[6] D. Bocka, C. Miller, and M. Ehmann, “Teaching and Learning Mathematics with Dynamic Worksheets”, International Journal of Continuing Engineering

Education and Life-Long Learning (IJCEELL), Volume 18, Issue 5/6, Inderscience Publishers, Geneva, 2008.

[7] N. Dai, L. Mandel, and A. Ryman, Eclipse Web Tools Platform. Developing Java Web Applications, Addison-Wesley Longman, Amsterdam, 2007.

[8] B. Eckel, Thinking in Java, Prentice Hall, Upper Saddle River (NJ), 2006.

[9] M. Ehmann, "Roboter zum Anfassen und virtuell – Problemlösendes Arbeiten im Informatikunterricht", in Spektrum – das Wissenschaftsmagazin der Universität Bayreuth, Universität Bayreuth, Bayreuth, 2006.

[10] D. Frayer and L. West, Creating a new world of learning possibilities through instructional technology, http://horizon.unc.edu/projects/monograph/CD/Instructional_Technology/Frayer.asp, last revisited 2009-05-11.

[11] S. Holzner, Eclipse, O'Reilly Media Inc., Sebastopol (CA), 2004.

[12] K.C. Hopson and S.E. Ingram, Developing Professional Java Applets, Sams Publishing, Indianapolis (IN), 1996.

[13] Internet Explorer, http://www.microsoft.com/windows/products/winfamily/ie/default.mspx, last revisited 2009-05-11.

[14] J. Kaplan, jvoicebridge, https://jvoicebridge.dev.java.net, last revisited 2009-05-11.

[15] A. Kay, "The Early History of Smalltalk", History of Programming Languages Conference (HOPL-II), Preprints, Cambridge (MA), 1993.

[16] J. Keogh and M. Giannini, OOP Demystified, McGraw-Hill/Osborne, Emeryville (CA), 2004.

[17] E. Klieme and J. Baumert, TIMSS – Impulse für Schule und Unterricht, Bundesministerium für Bildung und Forschung, Bonn, 2001.

[18] Massachusetts Institute of Technology – Media Lab – Lifelong Kindergarten Group, Scratch, http://scratch.mit.edu, last revisited 2009-05-11.

[19] Moodle, http://docs.moodle.org, last revisited 2009-05-11.

[20] OECD, PISA Results, OECD, 2000, 2003, 2006, http://www.oecd.org, last revisited 2009-05-11.

[21] B. Oberhaitzinger, H. Gerloni, H. Reiser, and J. Plate, Praxisbuch Sicherheit für Linux-Server und Netze, Hanser Fachbuchverlag, München, 2004.

[22] R. Palloff and K. Pratt, Lessons from the Cyberspace Classroom: The Realities of Online Teaching, Jossey-Bass, San Francisco (CA), 2001.

[23] W.H. Peterssen, Lehrbuch Allgemeine Didaktik, Ehrenwirth, München, 1983.

[24] Pong Arcade Game, http://en.wikipedia.org/wiki/Pong, last revisited 2009-05-11.

[25] M. Powell, jMonkey Engine User's Guide, http://www.jmonkeyengine.com/wiki/doku.php?id=user_s_guide, last revisited 2009-05-11.

[26] Qualifications and Curriculum Authority, The key skills qualifications standards and guidance, Qualifications and Curriculum Authority, London, 2004.

[27] Real VNC, http://www.realvnc.com, last revisited 2009-05-11.

[28] J. Russell, InnoSetup, http://www.jrsoftware.org/isinfo.php, last revisited 2009-05-11.

[29] A. Sikora, Technische Grundlagen der Rechnerkommunikation, Fachbuchverlag Leipzig, 2003.

[30] M. Spitzer, Lernen. Gehirnforschung und die Schule des Lebens, Spektrum Akademischer Verlag, Heidelberg, 2006.

[31] Squeakland, http://www.squeakland.org, last revisited 2009-05-11.

[32] Teamspeak, http://www.goteamspeak.com, last revisited 2009-05-11.

[33] V. Ulm, Objekte in Grafiken, Z-MNU Universität Bayreuth, Bayreuth, 2003.

[34] W3C, Architecure domain, Naming and Adressing: URIs, URLs, …, http://www.w3.org/Addressing/#rfc3986, last revisited 2009-05-11.

[35] WebDAV, http://www.webdav.org, last revisited 2009-05-11.

[36] Windows Meta File, http://www.microsoft.com/windows/windowsmedia/de/format/default.aspx, last revisited 2009-05-11.

[37] A. Zucker and R. Kozma, The Virtual High School: Teaching Generation V, Teachers College Press, New York (NY), 2003.

Distributed Emulator for Developing and Optimizing a Pedestrian Tracking System Using Active Tags

Junya NAKATA[*‡1] Razvan Beuran[*‡2] Tetsuya Kawakami[†3] Takashi Okada[‡*4] Ken-ichi Chinen[‡*5] Yasuo Tan[‡*6] Yoichi Shinoda[‡*7]
[*] Hokuriku Research Center, National Institute of Information and Communications Technology
2-12 Asahidai, Nomi, Ishikawa Japan
[1]jnakata@nict.go.jp [2]razvan@nict.go.jp
[†] Panasonic System Solutions Company, Panasonic Corporation.
4-3-1 Tsunashima-higashi, Kohoku, Yokohama, Kanagawa Japan
[3]kawakami.tetsu@jp.panasonic.com
[‡] Internet Research Center, Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa Japan
[4]tk-okada@jaist.ac.jp [5]k-chinen@jaist.ac.jp [6]ytan@jaist.ac.jp [7]shinoda@jaist.ac.jp

***Abstract*—In this paper we introduce a distributed emulator for a pedestrian tracking system using active tags that is currently being developed by the authors. The emulator works on StarBED, which is a network testbed consisting of hundreds of PCs connected to each other by Ethernet. The three major components of the emulator (the processor emulator of the active tag micro-controller, RUNE, and QOMET) are all implemented on StarBED. We present the structure of the emulator, how it functions, and the results from the emulation of the pedestrian tracking system. The emulator accomplished quite accurate emulation of ubiquitous network systems with the technique of emulation. We found several issues originated from active tag's firmware or protocol by applying the emulator to the emulation of the tracking system. We confirmed the results obtained by running tests corresponding to a real-world experiment.**

***Keywords*—ubiquitous networks; distributed testbed; supporting software**

I. Introduction

As Panasonic Corporation. (hereafter referred to as Panasonic) is developing a pedestrian tracking system using active tags, one requirement is to carry out a large number of trials. Real-world experiments with wireless network systems, and active tags in particular, are difficult to perform when the number of nodes involved is larger than a few devices. Problems such as battery life or undesired interferences often influence experimental results. We are currently implementing a solution by developing an emulation system for active tag applications that runs the real active tag firmware within a virtual, emulated environment. Through emulation, much of the uncertainties and irregularities of large real-world experiments are placed under control. In the same time, using the real active tag firmware in experiments enables us to evaluate exactly the same program that will be deployed on the real active tags; this is a significant advantage compared to simulation. For performing the practical experiments we use StarBED, a network experiment testbed.

StarBED consists of 920 PCs connected by two separated networks, the management network and the experiment network, as shown in Figure 1. StarBED provides a simulation supporting software, SpringOS, to implement an easy-to-use simulation environment with which the users can write experiment scenarios in a specific scripting language that can later be executed automatically. In order to be able to use this testbed for active tag emulation we developed several additional subsystems, and integrated them with the existing testbed infrastructure [1]. These subsystems were developed on the basis of existing tools that are already used on StarBED, namely the wireless network emulator QOMET [2], and the experiment support software RUNE [3].

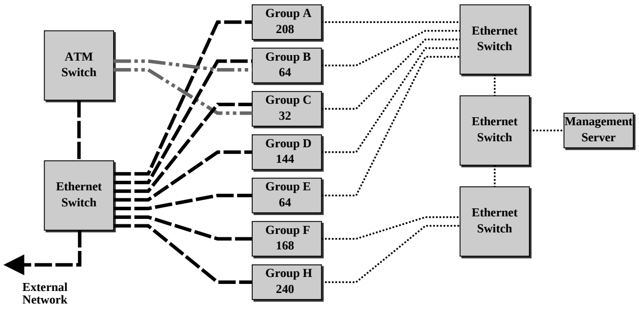


Fig. 1. Conceptual topology of StarBED.

Active tags were so far mainly studied through simulation, such as the work presented in [4]. Public domain wireless communication emulation research is currently mainly done in relation to Wireless LANs (WLANs). One can use real equipment, and hence be subject to potential undesired interferences. Two examples from this class that allow a controlled movement of wireless nodes are the dense-grid approach of ORBIT [5], or the more realistic robot-based

Mobile Emulab [6]. An alternative which avoids undesired interferences and side effects is to use computer models for real-time experiments. TWINE [7] is an example from this class. TWINE is a wireless emulator that combines wireless network emulation and simulation in one setup, but only supports 802.11b WLAN so far. Our development started from an existing wireless emulator, QOMET, which uses similar concepts.

There are already a number of implementations of experiment tools for ubiquitous systems that could be used in conjunction with active tag devices. Some of these tools focus on the operating system level, such as TOSSIM [8], which is a TinyOS simulator aiming to simulate TinyOS applications accurately in a virtual environment. ATEMU [9] is able to emulate TinyOS applications at processor level; its flexible architecture has support for other platforms too. ATEMU is thus closer to our purpose, since our low-cost active tags do not use any operating system. We aimed to run in emulation experiments the same firmware with the one used by the real devices. The manufacturer of the active tag processor , Microchip, only provides two alternatives for system development: real-time emulation in hardware using either the MPLAB REAL ICE In- Circuit Emulator, or the PICMASTER Emulator, or processor simulation using the MPLAB-SIM Simulator [10]. However none of these solutions are appropriate for our purpose; thus we developed our own real-time processor emulator running on PCs.

The pedestrian tracking system developed by Panasonic. makes use of active tags so as to provide to a central pedestrian localization engine the information needed to automatically calculate the trajectory to date and the current position of the active tag wearer. In the prototype system, three kinds of tags are used as shown in Figure 2. Mobile tags held by pedestrians transmit periodically ID packets which includes time information and sender's node ID. Fixed tags also transmit ID packets at certain intervals. Gateway tags have, in addition, a wired network connection to backend servers in which the data uploaded from mobile tags are sent. The trajectories of the pedestrians are calculated based on the data gathered by the gateway tags. Using the prototype of the pedestrian localization system, real-world experiments were carried out in March 2007. The experiment consisted in the orchestrated movement of 16 pedestrians both in indoor and outdoor environments. A system overview and experimental conditions will be presented later in this paper.

One of the important conclusions of the experiment was that it is very difficult to organize a real-world experiment for such applications of active tags. The number of people involved, and the accuracy of their movement following the predefined scenario, are only a few of the issues encountered. Nevertheless, the results of the above-mentioned experiment are currently being used as a basis for improving the prototype of the pedestrian localization system and extending it for use with very large groups of people, of the order of one thousand. The active tag emulation system that we designed and implemented plays an essential role at this point, since it makes it possible to continue the experiments in the development phase with ease and in a wide range of controllable conditions.

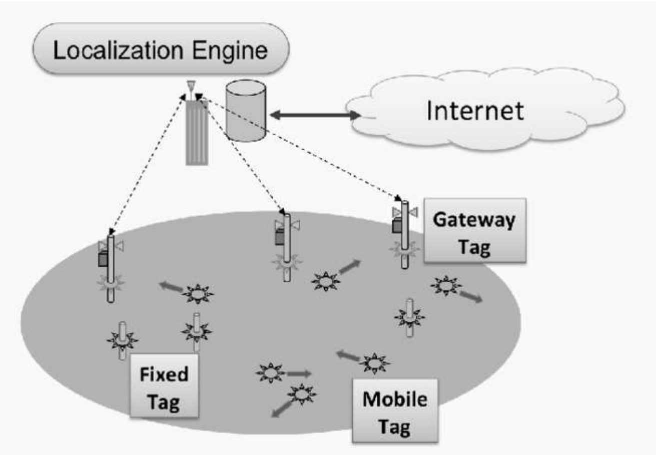


Fig. 2. Overview of the pedestrian tracking system.

The paper is organized as follows. Section II introduces our general approach to active tag emulation. This is followed by Sections III and IV, which describe the main components of the active tag emulation system: the wireless communication emulation, and the active tag processor emulation subsystems, respectively. Section V presents the preliminary real-world experiments carried out in order to validate the pedestrian localization system prototype. Section VI is dedicated to presenting experimental results obtained with the emulation system in the attempt to reproduce and extend the preliminary real-world experiments. The paper ends with sections on conclusions and references.

II. System Description

The technique of emulation implies creating a virtual environment in which the movement, the communication, and the behavior of active tags are all reproduced. Emulation has two main requirements in the case of our project: (i) Emulate in real time the wireless communication of the active tags; (ii) Emulate the active tag processor so that the same firmware used by the real devices can be tested in emulation experiments. The conclusions of the real-world experiment using the pedestrian localization prototype system were used as guidance during the design and implementation of the emulation testbed. In addition, we used our previous experience with emulation systems, such as those presented in [11] and [12], as we built the wireless communication emulation implementation on QOMET, as discussed in Section III.

The experiment-support software RUNE (Real-time Ubiquitous Network Emulation environment) is used to effectively run and manage the experiment in real time, as it can be seen in the overview given in Figure 3. RUNE Master and RUNE Manager are modules used in all RUNE-based experiments for controlling the experiment globally and locally, respectively. The active tag module was specifically designed and implemented for this application. This module includes: (i)Active

Tag Communication and chanel spaces, used to calculate and manage the communication conditions between active tags. These functions will be discussed in Section III; (ii)Active Tag Control space, which is powered by the active tag processor (PIC) emulator, and runs the active tag firmware in real time to reproduce the active tag behavior, as it will be discussed in Section IV. The experiment itself is performed using standard PCs (running the FreeBSD operating system) that are part of the StarBED testbed. They are labeled as Execution Units in Figure 3.

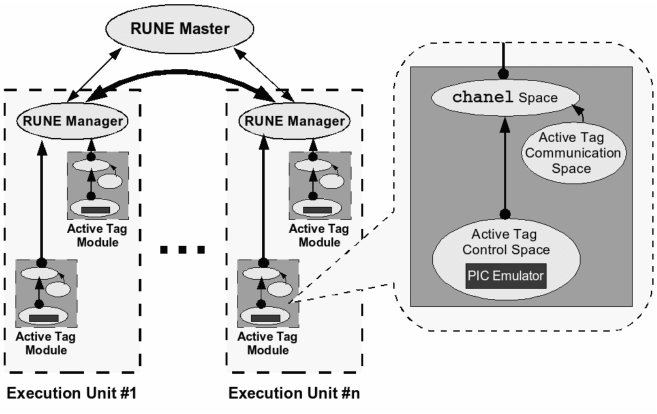


Fig. 3. Overview of the active tag emulation system.

The following RUNE configuration file is used for 16 pedestrian experiments:

```
#include "runebase.h"

BGNSPACELIST
  SPACE(mtag0,  xxx.yyy.zzz.2,   mtag.so)
  SPACE(mtag1,  xxx.yyy.zzz.3,   mtag.so)
                        ⋮
  SPACE(ftag0,  xxx.yyy.zzz.18,  ftag.so)
  SPACE(ftag1,  xxx.yyy.zzz.19,  ftag.so)
                        ⋮
  SPACE(gtag0,  xxx.yyy.zzz.22,  gtag.so)
  SPACE(gtag1,  xxx.yyy.zzz.23,  gtag.so)
                        ⋮
  SPACE(cspc0,  xxx.yyy.zzz.2,   cspc.so)
  SPACE(cspc1,  xxx.yyy.zzz.3,   cspc.so)
                        ⋮
ENDSPACELIST

BGNCONDUITLIST
  /* mtag0 */
  CONDUIT(mtag0,  cspc0)
  CONDUIT(cspc0,  mtag1)
  CONDUIT(cspc0,  mtag2)
                        ⋮
  /* mtag1 */
  CONDUIT(mtag1,  cspc1)
  CONDUIT(cspc1,  mtag2)
  CONDUIT(cspc1,  mtag3)
                        ⋮
ENDCONDUITLIST
```

III. QOMET

One of the most important elements when using emulation for studying systems that use wireless communication is to be able to recreate with sufficient realism the communication between them. For the active tags used in our pedestrian tracking system this was accomplished by extending the WLAN emulator QOMET to support the wireless transceiver used by active tags.

QOMET uses a scenario-driven architecture that has two stages. In the first stage, from a real-world scenario representation we create a network quality degradation (ΔQ) description which corresponds to the real-world events (see Figure 4).

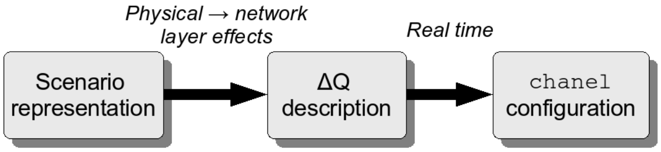


Fig. 4. Active tag communication emulation.

The ΔQ description represents the varying effects of the network on application traffic, and the wireless network emulator's function is to reproduce them.

The CHANel Emulation Library, chanel, is used to recreate scenario-specific communication conditions based on the ΔQ description (FER probabilities) computed by QOMET. Given that we emulate wireless networks, a second function of chanel is to make sure the data is communicated to all the systems that would receive it during the corresponding real-world scenario.

A. Active Tag Emulation

Our pedestrian tracking system uses the AYID32305 active tags from Ymatic Corporation., also known under the name S-NODE [13]. They were nicknamed communication tags or c-tags in the framework of the current pedestrian localization project. S-NODEs use as processing unit the PIC16LF627A microcontroller. The wireless transceiver of the active tag operates at 303.2MHz, and the data rate is 4800bps (Manchester encoding), which results in an effective data rate of 2400bps. The electric field emitted by active tags is 500μV/m; according to the specification, this produces an error-free communication range of 3-5m.

The active tag communication protocol was custom designed as a simple protocol based on time-division multiplexing. Each tag will select at random one of the available communication slots and advertise its identifier and the current time. Currently the number of available communication slots for advertisement messages is 9. There are additional communication slots that can be used on demand to transmit position tracking records from mobile tags to gateways.

The active tag communication model we currently use establishes the relationship between the distance between two nodes and the Frame Error Rate (FER, a data link layer parameter). This conversion is done based on measurements we made in an RF shielded room with the helicoidally shaped antenna, also used in the practical experiment, and 4-byte frames. By fitting a second degree equation on the measurement results we obtained the following equation:

$$FER_4\left(d\right) = 0.1096d^2 - 0.1758d + 0.0371 \quad , \qquad (1)$$

where FER_4 is the frame error rate (the index shows it is based on 4-byte frame measurements) and d is the distance between the receiver and transmitter active tags. The above equation gives a goodness-of-fit coefficient, R^2, equal to 0.9588.

In order to extend the communication range we introduced the constant C, the scaling factor, in equation III-A. Note that equation one needs some small modifications in order to represent accurately active tag communication range. The extended equation is:

$$FER_4'\left(d\right) = 0.1096\left(\frac{d}{C}\right)^2 - 0.1758\frac{d}{C} + 0.0371$$
$$FER_4\left(d\right) = \begin{cases} 0, & if\ \dfrac{d}{C} < 0.5m \\ 1, & if\ FER_4'\left(d\right) > 1 \\ FER_4'\left(d\right), & otherwise \end{cases} \quad . \qquad (2)$$

Since the measurements were done using 4 byte data frames, the result of equation (III-A) must be scaled accordingly for other frame sizes, as given by:

$$FER\left(d\right) = 1 - \left(1 - FER_4\left(d\right)\right)^{\frac{H+x}{H+4}} \quad , \qquad (3)$$

where FER represents the frame error rate for a data frame of x bytes, and H is the frame header size in bytes. In our pedestrian tracking system, x and H are constant, 7 and 6 respectively.

Slot collisions arising during the time-multiplexed communication are an additional and independent source of errors. However they are handled in real time during the live experiment in the receiving procedure of the processor emulator.

The frame error rate induced by slot collision, FER_s, is expressed by the equation below:

$$FER_s = \sum_{m=1}^{n} C_n^m / {N_{slots}}^{m+1} \quad , \qquad (4)$$

where C_n^m is the notation for combinations of a set of n objects taken m at a time, N_slots represents the number of slots used for communication (currently 9), and n is the number of c-tags transmitters that are located in the reception range of the current tag. For a sufficiently large number of slots, equation (4) can be simplified by ignoring the terms with $m > 2$, which become very small. In this case we obtain the following simplified relation:

$$FER_s = \frac{n}{{N_{slots}}^2} \quad . \qquad (5)$$

Considering that FER_x is equal to 1 for out of range transmitters, the number n can be computed at each moment of time as the cardinal of the set of c-tags, E, for which the frame error probability due to distance when received by the current tag is inferior to 1:

$$n = \left|E\right|, E = \left\{e|FER_x < 1\right\} \quad . \qquad (6)$$

Finally, the overall frame error rate, FER, can be computed by taking into account the fact that the two error causes discussed above are independent, as follows:

$$FER = FER_x + FER_s - FER_x \cdot FER_s \quad . \qquad (7)$$

A more realistic approach is to take into account the slot collision in real time during the live experiment in the receiving procedure of the PIC emulator. This approach required more computational power, and was used only selectively. If live slot collision emulation is enabled, than the model above needs to consider FER_s equal to 0.

B. Communication channel emulation for non-IP applications

Given that the active tags we emulate do not generate IP traffic, we could not use a wired-network emulator such as dummynet for introducing network layer effects to traffic, as previously done when using QOMET. As a consequence we decided to implement our own communication channel emulation system, named CHANEL (communication CHANnel Emulation Library). This module is inserted between the space emulating the c-tag (Active Tag Control Space in Figure 3) and its connection to the other spaces using conduits. The advantage of this integration is that it becomes transparent from the point of view of emulation whether RUNE spaces are executed on the same PC or on different PCs, since communication itself is handled transparently by RUNE conduits.

The main role of CHANEL is to recreate scenario-specific communication conditions based on the ΔQ description (FER probabilities) computed by QOMET. This function is similar to that of any wired-network emulator, such as dummynet. Given that we emulate wireless networks, a second function of CHANEL is to make sure the data is communicated to all the systems that would receive it during the corresponding real-world scenario. This is done by using the ΔQ description to decide the conditions for the communication between the current active tag and the other active tags in its transmission range. Since unicast-like traffic coming from an active tag needs to be sent to multiple destinations, there are concerns regarding the performance of CHANEL when the number of destinations increases. Note however that give the small transmission range of active tags (4-5m), the number of receivers that can be in the transmission range at one moment of time is relatively small. We estimate that in general the number will be of a couple of active tags, and may reach about 10 active tags when emulating crowded areas.

The communication channel emulation library was optimized to increase performance. In addition we now started using a binary file (the output of QOMET) inside CHANEL instead of the text file used so far. A main advantage is that the reduced size of the file allows for faster reading, and therefore improves CHANEL performance. Most delays that we measure reach occasionally values around 250ms.

Although we do not know exactly the source of these errors, we believe they are related to kernel scheduling parameters in FreeBSD. Note however that since we run the experiment ten times slower than real time, a time slot of 500ms has a length of 5s, therefore a 250ms configuration error only represents a 5% error.

As we want to be able to run multiple instances of CHANEL on the same computer, as well as provide a thread-safe environment, several mutex structures were added, and now concurrent access to CHANEL data structures became possible in a safe manner.

IV. Processor Emulator

One advantage of network emulation is that already-existing network applications can be studied through this approach to evaluate their performance characteristics. Although this is relatively easy for typical network applications that run on PCs, the task is complex when the network application runs on a special processor. In order to execute the active tag application unmodified on our system, we emulate the active tag processor so that the active tag firmware can be run in our emulated environment without any modification or recompilation.

Processor emulation in our system had to take into account the following aspects that we implemented:

(i) Instruction execution emulation; all 35 PIC instructions are supported by our processor emulator.

(ii) Data I/O emulation; the only I/O access method used by the active tag application is USART (Universal Synchronous Asynchronous Receiver Transmitter). The application uses USART to interface with the active tag transceiver, and also with the back-end system in the case of gateway tags.

(iii) Interrupt emulation; all interrupts necessary for the active tag application, i.e., timer0, timer1, and timer2 are supported.

We used a pseudo-DMA data transfer technique which is not implemented by the real device instead of emulating the active tag transceiver. It makes easier to integrate the active tag application and the peripheral components of the experiment such as the chanel space etc. We also used random number generation functionality to compensate the original active tag software's weakness in random number generation.

When emulating active tag applications such as ours it is important to introduce cycle-accurate processor emulation. In our case active tags use the time information contained in messages to synchronize with each others autonomously. Incorrect time information may lead to artificial desynchronization problems and potentially communication errors, therefore it must be avoided.

One of the main concerns regarding a processor emulator is how well the execution speed is reproduced, especially in the case when running multiple instances of the emulator. In Figure 5 we show how emulation accuracy changes depending on the operating frequency and the number of instances of the PIC emulator that are run in parallel. We remind that frequency used in the active tag application is 4MHz. The figure shows that, good accuracy is obtained for up to about 40 instances running in parallel when the operating frequency is 4MHz. We tried some scheduling algorithms such as Round Robin, EDF (Earliest Deadline First) etc. in order to obtain better performance. But no significant difference could not be seen because the scheduling of the processor instances takes place always in synchronous manner unlike the process scheduling of operating system.

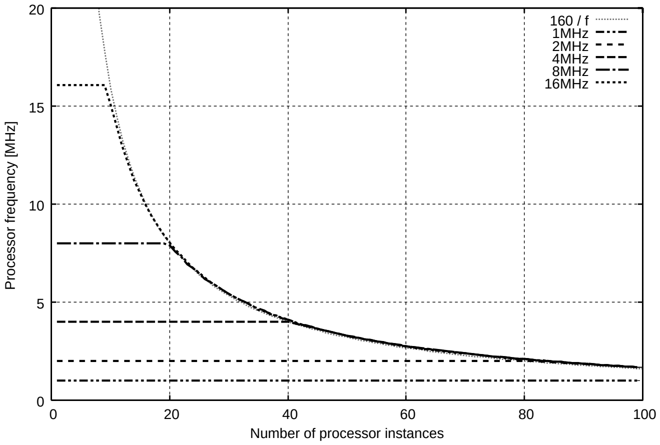


Fig. 5. Number of instances executed simultaneously at different frequencies on single PC

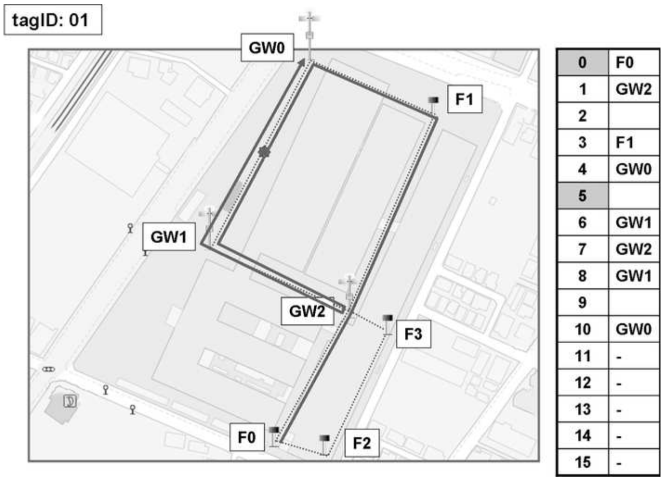
In our emulation, the PIC Emulator works as a part of Active Tag Control space as mentioned in Section 2. First of all, a PIC Emulator instance is allocated and initialized by invoking pic16f648Alloc(). The function allocates the data block for holding all processor internal states, registers, and memory and also launches the main emulation thread, which executes the fetch-decode-execute cycle repetitively. The main emulation thread controls the timing of progress of the emulation by using the RDTSC instruction of IA-32 architecture, which reads the Time Stamp Counter (TSC) register implemented in Intel IA-32 architecture processors. The advantage of this approach is: (i) The accuracy obtained in this way is theoretically the highest in a normal PC system, unless it has an external device which aids obtaining extremely accurate time such as GPS. (ii) It takes less time to execute the RDTSC instruction than typical C functions used to get system time, since the RDTSC instruction can be executed without the transition between kernel mode and user mode. There is also a thread created in the initialization process of the Active Tag Control space which takes care of the Pseudo-DMA data transfer. During emulation, both threads work together to accomplish real-time emulation of PIC processor.

V. Preliminary Trial

The real-world experiment was carried out in March 2007 by Panasonic. Each experiment participant was equipped with an active tag based pedestrian localization system prototype (c-tag).

A group of 16 participants were provided with instructions regarding the path they should follow in the 100 x 300m experiment area. An example of instructions, as received by participant #1 is shown in Figure 6.

The real-world experiment also included a number of tags with known position. These tags are divided into two classes: fixed and gateway c-tags, denoted in Figure 6 by F0 to F3, and GW0 to GW2, respectively. The role of fixed tags is to provide specific information to the mobile ctags that come in their vicinity to makes it possible to localize those tags. Gateway c-tags, in addition to c-tag communication, also allow information to be transferred between them and to the back end system. The gateways are placed at 3 known outdoor locations Gateways are also connected to the back-end servers; their data is used by the localization engine to determine the trajectories and positions of pedestrians.



0	F0
1	GW2
2	
3	F1
4	GW0
5	
6	GW1
7	GW2
8	GW1
9	
10	GW0
11	-
12	-
13	-
14	-
15	-

Fig. 6. Pedestrian movement instructions as received by participant #1.

The real-world experiment was successful in the sense that data collected from the active tags could be used to localize the pedestrians in most cases with sufficient accuracy. The active tag localization approach doesn't use any GPS-like or triangulation system. Instead the logs of each mobile tag, as collected by gateways, are used. The c-tag logs contain information regarding the time at which other mobile or known-position c-tags were encountered, and their identifiers. This information is used to predict the trajectory of c-tag wearers and track their position. The basic equation used to calculate the position Px of a pedestrian at moment of time tx is:

$$P_x = P_i + (P_j - P_i)\frac{t_x - t_i}{t_j - t_i}, \tag{8}$$

where P_i and P_j are the known positions of the pedestrian (from ctag logs) at moments of time t_i and t_j, with $t_i \leq t_x \leq t_j$. For more details about the experiment and the pedestrian localization engine one may consult [14] (in Japanese).

VI. Results

The emulation shown uses exactly the same conditions as the real-world experiment described in Section V, and was used to validate the emulation system. For simplicity each active tag and the associated chanel component are run on one PC. The emulational setup follows the overview presented in Figure 3.

A. Emulation results (16 virtual pedestrians)

The initial position of the 16 virtual pedestrians, the locations of the 4 fixed c-tags and 3 gateway c-tags, the building topology, and virtual pedestrian movement were all described by converting the real-world experiment instructions to the QOMET XML-based scenario description. Time granularity used when computing communication conditions, as well as during real-time execution was 0.5s. RUNE was used to configure the host PCs according to the emulation description and run the emulation.

We implemented a tool which converts the result of the emulation into KML format [15]. Visualizing the result with Google Earth^{TM} [16] (Figure 7) [1] helps to figure out the motion of the virtual pedestrians in time and to easily identify localization problems.

In order to understand better the localization errors, it is possible to draw for each virtual pedestrian its emulated trajectory, and the trajectory localized by the system. By comparing them, and seeing where differences occur, one can determine where the algorithm needs to be improved.

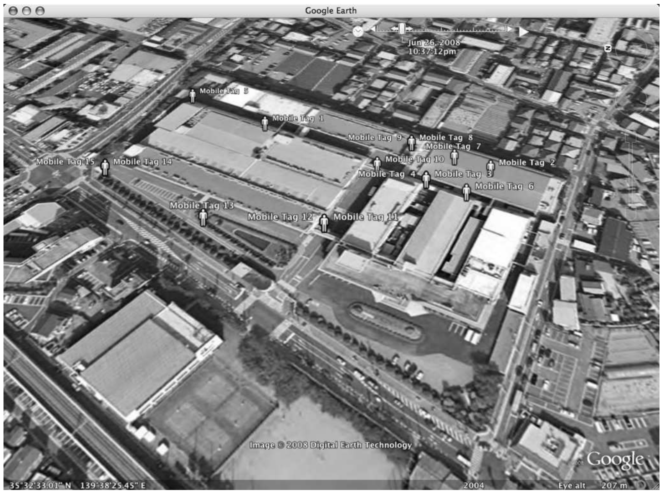


Fig. 7. Visualization using Google Earth^{TM}.

We have performed several series of emulations trying to reproduce and extend the 16 virtual pedestrian experiment carried out by Panasonic. The communication range of active tags is one of the most important parameters, since it determines the area in which communication is possible. Communication range is given by transmitted power; therefore it is directly related to power consumption. In a series of emulations,

[1] Google Earth^{TM}mapping service is a trademark of Google Inc.

we tried to see what is the performance of the localization algorithm for several communication ranges, as follows: 3m, 6m, 9m, 12m, 15m (see Figures 8 and 9). By looking at the mean localization error versus range in Figure 9, one can conclude that the range of 9 m seems to provide optimum performance in this case.

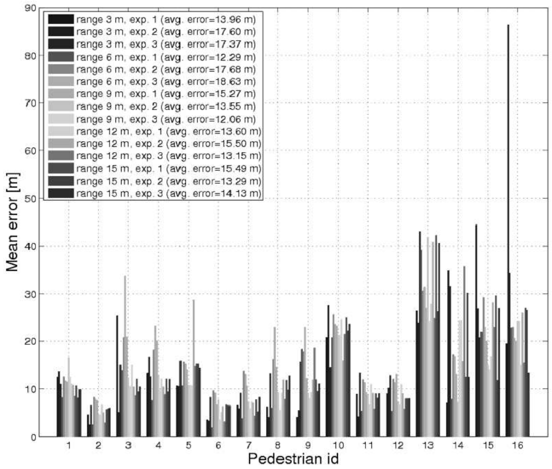


Fig. 8.   Mean localization error per virtual pedestrian for several communication ranges (3 emulations per range).

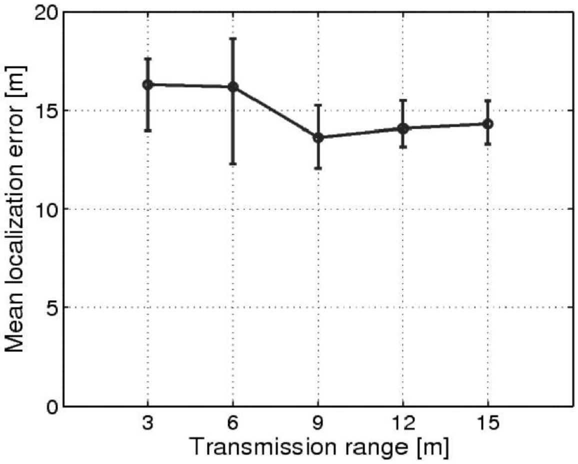


Fig. 9.   Mean localization error versus communication range.

Emulation can be used to investigate a wide range of controllable conditions. We decided to use this approach to determine how localization performance changes when the number of slots allocated to communication between tags varies. For this purpose we performed several emulations, both with the tags configured for 3m range, and 9m range. In each series of emulation we varied the number of slots as follows: 3, 6, 9. Note that 9 slots is the value used by the real prototype. The results are shown in Figures 10 and 11. Analyzing the mean localization error versus the number of slots in Figure 11, we conclude that for 9 m range using 6 slots is enough, but the 3 m range does require the use of 9 slots to provide best performance.

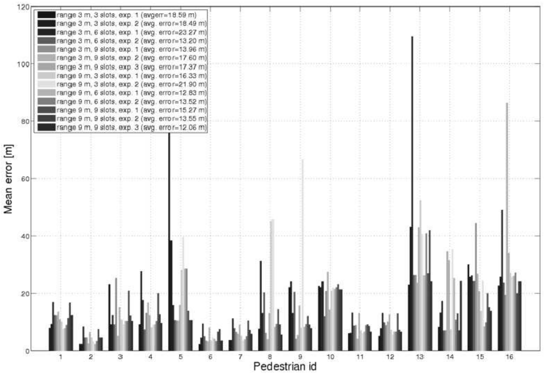


Fig. 10.   Mean error per virtual pedestrian when varying the communication range and the number of communication slots used by each tag.

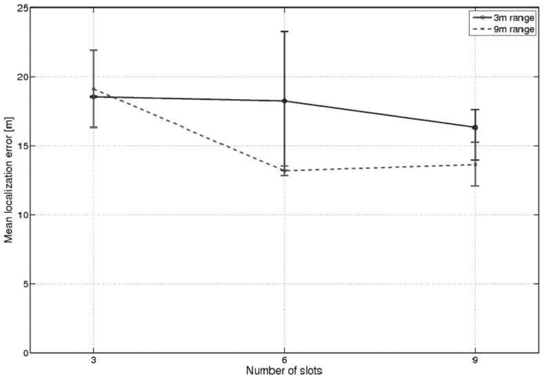


Fig. 11.   The mean localization error per emulation versus the number of communication slots.

In Figures 12 and 13 we show the results for another emulation in which we configured the range of the active tags to 5m, 10m and 15m. The purpose was to demonstrate how our system could be used to determine the optimum range for the active tags.

Figure 13 indicates that for a range of 10m, the optimum performance is achieved. For 5m range, the variation of the error is quite high, since in some case the short communication range leads to the event that two tags may miss the chance to communicate, therefore their trajectory may not be correctly identified. On the other hand for communication range of 15m, the localization error increases slightly. This is explained by the fact that with a longer range the communication between tags starts and lasts until a longer distance, therefore the accuracy of localization decreases.

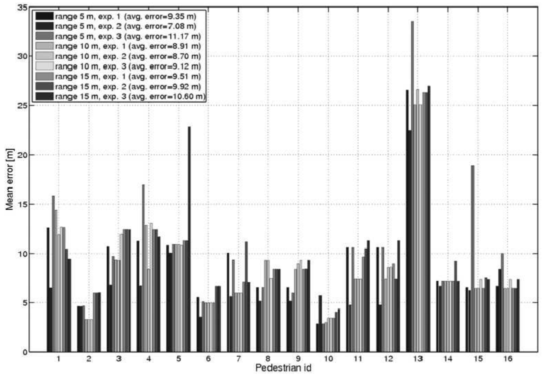


Fig. 12. Mean error for each virtual pedestrian for several communication ranges (3 emulations per range).

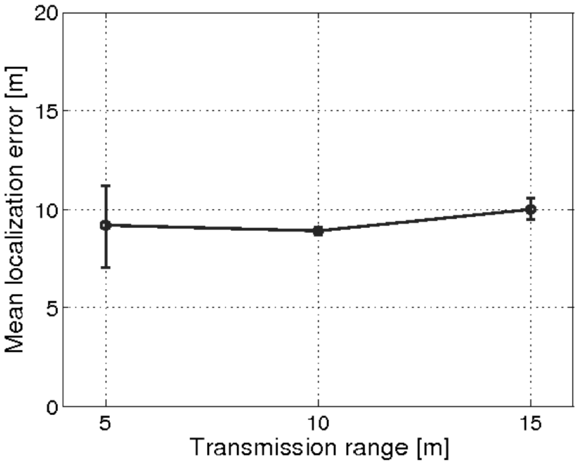


Fig. 13. Mean error for each transmission range.

In Figure 14, we show the visualization tool we use for the communication protocol of the active tags. Such a graphical representation gives an insight in the timing of the messages sent and received by active tags, as well as other elements of the communication protocol. This tool was successfully used to identify some potential firmware implementation problems. For instance, a weakness of the random number generator implementation led to the choice of the same time slot for communication in our emultaions. This fact produced an unusually large number of collision effects, for which the cause became obvious using the communication visualizer tool. As mentioned in Section IV, we implemented an alternative random number generator in the PIC emulator as a temporary solution. The implementation of the random number generation functionality is planned to be improved in the next prototype localization system.

Another issue we were able to identify by emultaions is related to time synchronization between active tags. At the moment a mobile active tag synchronizes its clock based on the time received from neighboring tags. Gateways and fixed nodes do not synchronize their time. We observed in our emulation system that the time accuracy without time synchronization (e.g., for gateways) is better than with time synchronization (i.e., for mobile tags). The time drift of two or more mobile nodes that are not in the vicinity of a gateway or fixed tag becomes quickly significant using the current synchronization algorithm, while the gateways and fixed tags themselves seem to be relatively stable, although not using time synchronization. This issue had not been noticed in the real-world experiment, but it is very important. A significant time drift leads to localization inaccuracy and must be solved in the next prototype. We circled in Figure 14 an example of time drift for a pair of mobile tags (P10 and P11).

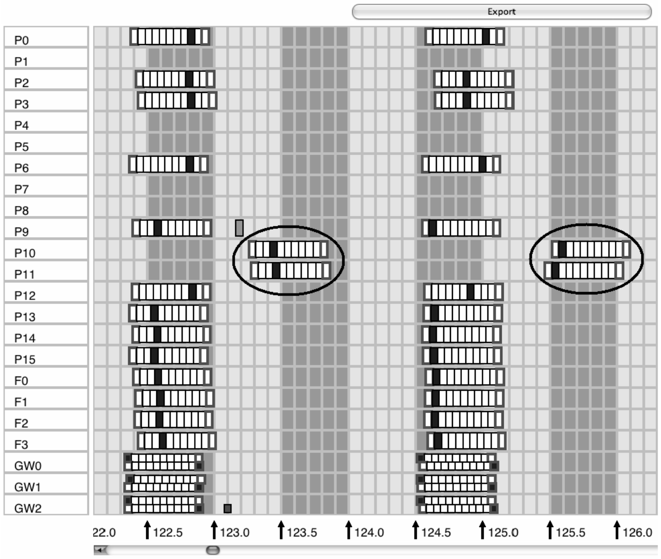


Fig. 14. Active tag communication visualization tool.

Figure 15 shows at where the #1 tag exchanged packets that used for localization to other tags. As the figure shows, enough number of packets necessary for localization were exchanged in our emulation. All the result presented in this section indicates the emulated tag software works properly even though we, unfortunately, have no way to confirm if the behavior of emulated tag software is correct by comparing with the result obtained from the real-world experiment since the real tags does not have any logging functions due to memory and processing ability restrictions.

B. Emulation results (100 virtual pedestrians)

One of the purposes of developing emulation was running large-scale emulation that cannot be executed very easily in the real world. In this content we ran several emulations with up to 100 virtual pedestrians. For the emulations, the motion of virtual pedestrians is generated by a motion generator using the real geographical information provided by GSI (Geographical Survey Institute, a Japanese governmental organization) as the constraint condition. Figure 16 shows an example of generated trajectory of virtual pedestrians.

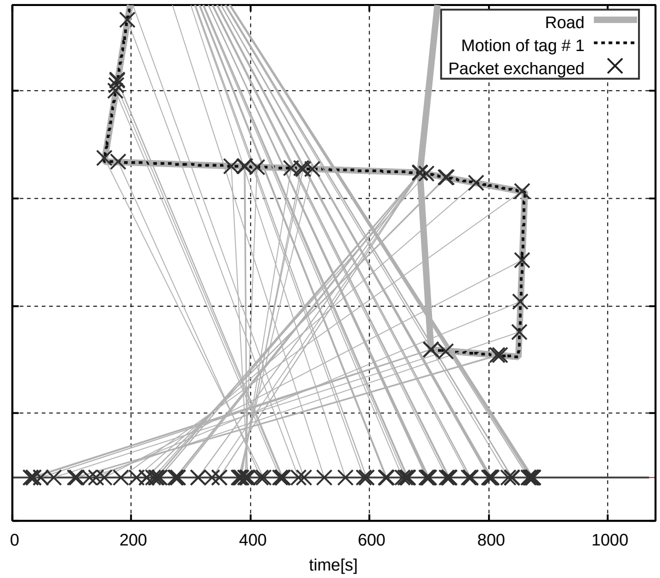


Fig. 15. Packet exchanged location and trajectory of the #1 tag.

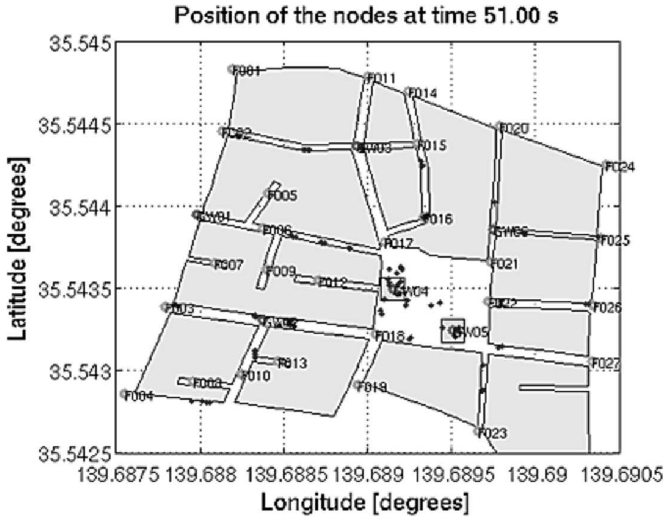


Fig. 16. Generated topology of 100 virtual pedestrian emulation.

Although we performed several series of 100 virtual pedestrian emulations, due to some problems in the localization engine we are not able to plot the localization results in the same way we did for the other emulation. The problem was the following: in the 100 virtual pedestrian emulation, the number of tags that reaches the destinations at GW4 and GW5 is high, therefore there are many collisions between the mobile tags as they try to upload their information. In addition, due to the relatively big size of the area, the number of encounters between mobile tags is rather small. These two reasons lead to the fact that not all the tags manage to upload information to gateways. Table I illustrates how many mobile tags never succeeded to upload information during the emulation. The table shows that almost half of the mobile tags never succeeded to upload any information although the rest of tags uploaded hundreds of packets as Figure 17 shows. The current version of the localization engine is not able to cope with the case when incomplete information is given, and was not able to produce any results. Panasonic is currently investigating this issue so that the robustness of the localization engine can be increased.

TABLE I
NUMBER OF TAGS SUCCEEDED TO UPLOAD INFORMATION.

Number of tags that uploaded at least one record	Number of tags that uploaded no record
55	45

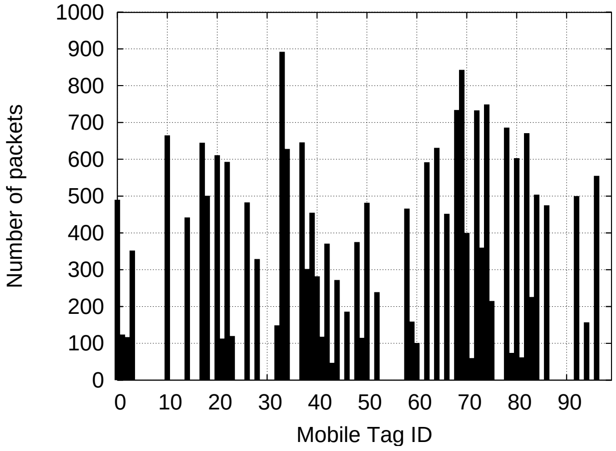


Fig. 17. Number of P records uploaded by each mobile tag.

Table II shows how many records are left in the memory of each mobile tag at the end of emulation. According to the table, over 90% of mobile tags had had information not uploaded at the end of emulation. This happens if a mobile tag lost the opportunity to upload information in the final part of its trajectory for some reasons, mainly collision in the emulation. So Panasonic is now designing a collision avoidance protocol, since our emulations have shown that without such an algorithm the active tag localization system cannot function for relatively crowded areas. This is one of the important findings of our emulations.

TABLE II
NUMBER OF TAGS HAVE INFORMATION LEFT IN MEMORY.

Number of tags have information left in memory	Number of tags have no information left in memory
91	9

VII. CONCLUSION AND FUTURE WORK

In this paper we presented an emulation system that we designed and developed for active tag applications. This emulation system is currently employed for the development phase emulations of a pedestrian localization system by Panasonic.

By using our system it was possible to simplify the development and testing procedures of the localization engine, and identify several firmware implementation issues.

In order to validate the emulation system we carried out tests that reproduced a real-world 16 pedestrian experiment that took place in March 2007 using the prototype of the active tag based pedestrian localization system. The emulation results show the good agreement that exists between the virtual motion patterns of pedestrians, reproduced according to the real-world scenario, and the actual conditions that were recreated in our emulation.

Through emulations we found several issues originated from active tag's firmware or protocol. Some of the issues such as those of time synchronization and random number generation are already fixed by modifying firmware in emulation first and then feeding it back to the real firmware. Some more fundamental issues such as unreliable behavior in crowded areas are under study by Panasonic and supposed to be fixed in the next version of the firmware. In both cases, our distributed emulation approach utilizing the real firmware made it easy to find out problems and fix them. This fact tells that the emulation environment implemented on distributed environment is useful to validate systems especially which consists of many small components such we targeted.

Our future work has several main directions: improve the scalability of the system so as to enable emulations of pedestrian groups as large as 1000; improve the realism of the wireless communication emulation by using more accurate 3D models for topology and electromagnetic wave propagation; combine the behavioral motion model with a GIS-based urban area description to create a realistic pedestrian trajectory generator for large-scale urban emulations.

REFERENCES

[1] Junya NAKATA, Razvan Beuran, Tetsuya Kawakami, Ken ichi Chinen, Yasuo Tan, and Yoichi Shinoda. Distributed emulator for a pedestrian tracking system using active tags. In *UBICOMM 2008: Proceedings of the 2008 The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, pages 219–224, Washington, DC, USA, 2008. IEEE Computer Society.

[2] Razvan Beuran, Lan Tien Nguyen, Khin Thida Latt, Junya Nakata, and Yoichi Shinoda. Qomet: A versatile wlan emulator. In *AINA '07: Proceedings of the 21st International Conference on Advanced Networking and Applications*, pages 348–353, Washington, DC, USA, 2007. IEEE Computer Society.

[3] J. NAKATA, T. Miyachi, R. Beuran, K. Chinen, S. Uda, K. Masui, Y. Tan, and Y. Shinoda. Starbed2: Large-scale, realistic and real-time testbed for ubiquitous networks. In *The 3rd International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities(TridentCom 2007), Orlando, Florida, U.S.A.*, 2007.

[4] A. Janek, C. Trummer, C. Steger, R. Weiss, J. Preishuber-Pfluegl, and M. Pistauer. Simulation based verification of energy storage architectures for higher class tags supported by energy harvesting devices. volume 32, pages 330–339, Amsterdam, The Netherlands, The Netherlands, 2008. Elsevier Science Publishers B. V.

[5] Rutgers University and Wireless Information Network Laboratory. *ORBIT - Wireless Network Testbed*. http://www.orbit-lab.org/ 15.05.2009.

[6] David Johnson, Tim Stack, Russ Fish, Daniel Montrallo Flickinger, Leigh Stoller, Robert Ricci, and Jay Lepreau. Mobile emulab: A robotic wireless and sensor network testbed. In *INFOCOM*. IEEE, 2006.

[7] Junlan Zhou, Zhengrong Ji, and Rajive Bagrodia. Twine: A hybrid emulation testbed for wireless networks and applications. In *INFOCOM*. IEEE, 2006.

[8] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. Tossim: accurate and scalable simulation of entire tinyos applications. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 126–137, New York, NY, USA, 2003. ACM.

[9] J. Polley, D. Blazakis, J. McGee, D. Rusk, and J. S. Baras. Atemu: A fine-grained sensor network simulator. In *Proc. of the First IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON 2004), Santa Clara, California, U.S.A.*, 2004.

[10] Microchip Technology Inc. *MPLAB*. http://www.microchip.com/ 15.05.2009.

[11] R. Beuran, J. NAKATA, T. Okada, T. Miyachi, K. Chinen, Y. Tan, and Y. Shinoda. Performance assessment of ubiquitous networked systems. In *5th International Conference on Smart Homes and Health Telematics (ICOST2007), Nara, Japan*, pages 19–26, 2007.

[12] Takahashi Okada, Razvan Beuran, Junya Nakata, Yasuo Tan, and Yoichi Shinoda. Collaborative motion planning of autonomous robots. volume 0, pages 328–335, Los Alamitos, CA, USA, 2007. IEEE Computer Society.

[13] Ymatic Inc. *S-NODE specification*. http://www.ymatic.co.jp/ 15.05.2009.

[14] Y. Suzuki, T. Kawakami, M. Yokobori, and K. Miyamoto. A real-space network using bi-directional communication tags - pedestrian localization technique and prototype evaluation. In *IEICE Forum on Ubiquitous and Sensor Networks, techni cal report*, 2007.

[15] Open Geospatial Inc. *KML Standard*. http://www.opengeospatial.org/standards/kml/ 15.05.2009.

[16] Google Inc. *Google Earth*. http://earth.google.com/ 15.05.2009.

A Home Context-Aware System with a Mechanism for Personalization of Service Providing

Hiroyuki Yamahara, Takanori Soma, Fumiko Harada, Hideyuki Takada
Ritsumeikan University
1-1-1 Noji-Higashi, Kusatsu, Shiga 525-8577, Japan
{yama, takanori}@de.is.ritsumei.ac.jp, {harada, htakada}@cs.ritsumei.ac.jp

Yukihiro Shimada
GOV Co., Ltd.
134 Minami-cho, Tyudoji, Shimogyo-ku, Kyoto 600-8813, Japan
shimada@go-v.co.jp

Hiromitsu Shimakawa
Ritsumeikan University
1-1-1 Noji-Higashi, Kusatsu, Shiga 525-8577, Japan
simakawa@cs.ritsumei.ac.jp

Abstract

We propose a home context-aware system which has a mechanism for personalization of service activation and context estimation. Personalization of service activation is to realize service activation along user intention. The proposed system can activate a variety of services along user intention by combining a system-active approach and a user-active approach. Because users choose activated services finally, the system can activate services even along user intention which cannot be inferred by computers. Personalization of context estimation is to realize setting values appropriate for each user to parameters used for context estimation in a system-active approach. The proposed system determines values appropriate for each user by utilizing statistical data of test users whose characteristics are similar to each user. Determination of individual values enables stabler context estimation than context estimation with values common to all users.

Keywords - home intelligent service; context awareness; RFID; threshold; behavior

1. Introduction

People would make mistakes in daily activities in homes. They sometimes leave their homes without closing windows and turning off an air conditioner. They go to bed without locking the front door. We are developing a home context-aware system which provides services to prevent users from facing dangers of making mistakes in homes and to make their life more comfortable. Before the users leave their homes, our system can warn them to lock windows and to turn off the air conditioner. The system can also lock windows and turn off the air conditioner instead of them. There have been already technologies which lock doors of house by remote control with a cell phone. However, these technologies cannot prevent dangers, because these technologies are user-active approaches and can be useful only in a case the users get aware of their mistakes by themselves. Moreover, it is difficult for users unfamiliar with computers, such as elderly people, to utilize their cell phones actively.

As an approach for solving these problems, there are studies of context-aware systems which provide services according to user context such as user behavior and user situation. Because the systems estimate user context, the systems can awaken users to something they are not aware of. In addition, because services are provided with a system-active approach, it can provide services to users unfamiliar with computers without any active operations of users themselves. Meanwhile, it is impossible for the systems to estimate every user context accurately. Therefore, this approach is at risk for providing services inappropriate for user situation because of misestimation by the systems. Existing studies of context-aware systems focus on improvement of estimation accuracy of user context, but they do not consider handling of inappropriate services provided based on misestimation.

Not to make users feel dissatisfied, home context-aware systems must have a mechanism for preventing inappropriate service providing based on misestimation. Moreover, the systems are expected to have a mechanism for activating a variety of services along user intention which cannot be inferred by computers. For example, the systems can infer a user leaves his home but cannot infer whether he goes away for a long time or for a short time. If he goes away for a long time, a service for turning off lights and air conditioners is useful. If he goes away for a short time, he may prefer a service for turning off lights but leaving air conditioners as it is. In this way, there can be sometimes different services according to detailed user intention, which cannot be inferred, even in a scene of leaving home. Useful systems should be able to activate a variety of services, which have different details even in a same scene, along detailed user intention. In this paper, to realize such service activation along user intention is referred to as personalization of service activation.

In addition, because home context-aware systems can be introduced into a variety of users who have different characteristics of behavior, context estimation methods of the systems must be stably effective for not some users but as many users as possible. Appropriate values of parameters used as criteria of judgment in the estimation methods vary among users. If values common to all users are used, estimation accuracies on some users become significantly low. It is preferable to determine the values of parameters individually. Values appropriate for each user can be determined by collecting sensor data acquired according to his daily activities for a long period and analyzing the data. However, service providing should be early started after introducing the systems into homes. Therefore, the values must be determined with a small number of data of each user which can be collected in a short period. In this paper, to realize setting values appropriate for each user to parameters used for context estimation is referred to as personalization of context estimation.

The system we previously presented[18] considers a mechanism for preventing inappropriate service providing, but does not enough consider a mechanism for these personalization. In this paper, we propose a home context-aware system with a mechanism for personalization of service activation and personalization of context estimation, which has been improved from the previous system.

In our system, short-range passive RFID tags are installed in a variety of objects such as a doorknob, a wallet, a wristwatch, a refrigerator in homes. A unique ID is stored in each tag. The user wears a ring-type RFID reader on his hand. Using this RFID system, the histories of objects touched by the user in his home are stored in a home server.

The proposed system personalizes service activation by combining a system-active approach and a user-active approach. Services are provided in this system as follows. First, specific user behavior is detected from kinds of touched objects and the order of the objects. Next, with the detection as a trigger, service candidates according to the detected user behavior are narrowed down and they are offered the user. At the same time, the service candidates are respectively mapped to objects around the user. These objects become switches for activating the offered services temporarily. The user can choose the objects mapped with service candidates from objects around him by himself. Finally, services are activated just by user's touching to these objects. In this way, the proposed system provides services along user intention by choosing activated services with a user-active approach from service candidates chosen with a system-active approach.

In addition, the proposed system has a mechanism for determining initial values of individual users for parameters used in a behavior detection method as a mechanism for personalization of context estimation in the system. The system utilizes statistical data on test users, which are acquired before introducing the system into the home of each user. Suppose the system is introduced into a user υ. First, a small number of data of υ is collected in a short period at the initial stage after introducing the system. Next, the data of υ is compared with the data of test users, based on the concept of collaborative filtering. Finally, values appropriate for υ are determined from data of test users who have characteristics similar to υ.

The proposed system which has a mechanism for personalization of service activation and context estimation has the following advantages.

- By combining a system-active approach and a user-active approach, the system can activate a variety of services along user intention without losing an advantage that the system can awaken users to something they are not aware of.
- Determination of individual initial values for parameters in the detection method enables stable behavior detection by improving detection accuracies of users whose detection accuracies are low with values common to all users.

The remaining part of this paper is organized as follows. Section 2 presents problems on personalization of service activation, and Section 3 presents problems on personalization of context estimation in the proposed system. Section 4 shows the proposed system has a mechanism for personalization of service activation, with the flow of service providing in the system. Section 5 describes a method for determining individual values of parameters in the behavior detection method as a mechanism for personalization of context estimation in our system. Section 6 presents an experimental life space where the proposed system is implemented. In Section 7, we evaluate the usability of the touch-to-object interface for service activation to consider the possibility of a user-active approach which is a core of personalization of service activation. In Section 8, we evaluate accuracy of behavior detection and personalization of context estimation with the proposed system. Section 9 presents challenges for improvement of our system. Finally, we conclude this paper.

2. Problems on Service Activation

2.1. User's Final Decision of Services

The system for supporting daily life is required not to take actions which are against user intention because such actions make discontent. Therefore, it is necessary to improve the accuracy of context estimation for reducing inappropriate services caused by false estimation. However, it is practically difficult to achieve 100 percent accurate estimation along user intention at any given time with computers. Reducing inappropriate services is possible but it is impossible to eliminate such services completely. For example, suppose the system provides a service for turning off lights which a user forgets to turn off before he leaves his home. In a case he goes away for a long time, he wants to turn off lights. But in the case he goes away for a short time or a case a housemate is in his home, he may not want to turn off lights. It is not easy to discriminate such cases whose details are different. To provide services which are along user intention even in such cases, it is preferable that the user finally decides whether or not services should be activated. At the same time, the interface he uses to decide whether he activates services or not must not be complicated. It must be simple and costless so that users unfamiliar with computers can use intuitively without being conscious of computers.

2.2. Service Activation on the Spot

The interface which users use to decide activated services should be available regardless of user position. Suppose the system warns a user that lights in the living room are still on, after putting on his shoes on the front door when leaving his home. It is annoying for the user to go back to the living room, where a home server is located, to send a command for activating a service which turns off the lights to the system. He may rather turn off the lights directly by himself in the living room than activate the service after moving from the front door to the living room. The usefulness of the system which can be used only on a specified place is low. Users should be able to activate services on the spot according to their position.

2.3. Related Works on Service Activation

There are studies of easy-to-use interfaces to activate services with operation of users themselves without automatic activation of systems according to user context.

Nichols et al. have improved the interface of a cell phone used for remote control of home appliances [10]. However, it is difficult for users unfamiliar with computers to operate a cell phone. Tsukada et al. propose remote operation with finger gesture [16]. Although this interface does not make users conscious of computers, users may waste a long time to learn how to operate it because it is not intuitive. Riekki et al. propose an interface for activating services with RFID tags attached by symbol images, with which users can intuitively know the content of activated services [14]. However, users cannot activate services without moving to specified places to touch specified tags, because tags are fixed on specified places and services are fixed on the tags. In addition, although technologies of speech recognition are studied as an interface to activate services with speech of users [4], the technologies are not enough practical at present. More comfortable interfaces are necessary to activate services by users themselves.

3. Problems on Behavior Detection

3.1. Complexity of User Behavior

We address behaviors which can be triggers to provide services as user behavior in this paper. They are detected by observing a sequence of habitual activities of individual user. Suppose the user brushes his teeth, goes to the toilet, picks up a wallet, wears a wristwatch and opens the front door in order. By observing such a sequence of some habitual activities which are taken before leaving home, his leaving home can be detected. Suppose he opens the front door. It is difficult to judge whether he leaves his home only from one activity. He may go out for picking up a newspaper. Similarly in another example, his getting up can be detected by observing a sequence of activities taken right after getting up, such as he goes out of bed, stops an alarm clock, turns on lights and drinks water from the faucet. Note that kinds of habitual activities and habitual order of them depend on individual user.

The user does not always take same activities in same order every time. In the observed sequence of user activities, habitual order relation and non-habitual order relation are mixed. For example, before leaving home, there can be habitual order from "going to the toilet" to "picking up a wallet" and from "picking up a wallet" to "opening the front door" but there may be no habitual order relation between "picking up a wallet" and "wearing a wristwatch". In addition, sometimes rare activities such as "picking up an umbrella" in a rainy day are inserted into the activity sequence. By contraries, part of habitual activities may be also sometimes omitted. From such a complex sequence, it is necessary to extract characteristic kinds and order which represent individual habits indicating user behavior to achieve behavior detection.

Some existing studies propose methods for recognizing user motion such as "walking" and "standing up"[2, 9]. Other studies propose methods for recognizing simple activities such as "brushing teeth", whose characteristics are similar among users[12, 13, 17]. User behaviors as triggers of service providing, such as "leaving home", need to be detected by observing a sequence of such activities. Logan et al.[8] and Huỳnh et al.[5] study methods for recognizing behaviors such as "leaving home" called as $high\text{-}level$ $activities$. Basically, these existing methods aim to achieve classification or labeling of activities to identify user activity on a certain period of time. Compared with these, behaviors in this paper are not recognized but should be detected as triggers of service providing. Because the behavior handling considered in this paper is handling which is reactive to user behavior, it is a different target from existing methods.

3.2. Deadline for Providing Services

Some services have definite deadlines of providing them, while others have no definite deadline of providing them.

Examples of the former are services provided when the user leaves his home or he goes to bed. Suppose he is warned that he does not have wallet after he leaves his home. He must go back to inside his house to pick up his wallet. The value of the service provided after leaving home is significantly lower than that before his leaving. The deadline of providing services is the instant the user goes out of his house through the front door. Similarly, suppose the user is warned that the front door is not locked when he goes to bed. The deadline of such services provided on going to bed is the instant he goes sleep in his bed. To provide high value services for the user, his behaviors such as leaving home and going to bed must be detected before the deadline.

Examples of the latter are reminder services provided when the user gets up or comes home. Suppose the service, which reminds him one day schedule and what to do on the day, is provided with detection of his getting up as a trigger. Such service can prevent his mistakes. Such reminder services triggered by detection of getting up or coming home have no definite deadline of providing them, nonetheless it is preferable to provide the services within the time the user is doing a series of activities right after getting up or coming home.

Behaviors which can be triggers of service providing must be detected before the deadlines of services.

3.3. Template Matching

In home context-aware systems, data of user behavior is acquired online from sensors. This paper refers to the sensor data as $behavior\ log$. Generally, behavior of a user is inferred by template matching with behavior logs. The flow of template matching is as follows.

1. A certain amount of behavior logs of the user are collected.

2. A template which represents characteristics of user behavior is created by statistical analysis of the collected behavior logs. The behavior logs used for creating the template are referred to as $sample\ behavior\ logs$. The created template is referred to as a $matching$ $template$.

3. User behavior is inferred by checking the degree of conformity when matching current behavior logs, which are acquired online from sensors, with the matching template. Matching is repeated at some kind of specified timing. The behavior logs which are matched with the matching template is referred to as $match\text{-}target\ behavior\ logs$.

In our system, user behavior is detected based on template matching. The system cannot detect user behavior until the matching template is created after introducing the system into homes. That means users are not provided services. If it takes a long time to create the matching template, users are dissatisfied with waiting for a long time until the start of service providing. Sample behavior logs of individual user need to be collected in a short period to start providing services early. That is, the system should create an initial matching template with a small number of individual sample behavior logs which can be collected in a short period to be accepted by users.

Behavior logs are classified into two types to a matching template. There are $true\ cases$ and $false\ cases$. Suppose there is a matching template to detect a behavior of leaving home. True cases are behavior logs in scenes where a user is leaving home. False cases are behavior logs in other scenes.

3.4. Setting of Initial Threshold Values

Some parameters should be determined appropriately in a method for behavior detection to achieve high detection accuracy. In particular, the following two threshold values have an impact on the detection accuracy. One is a threshold value used for creating a matching template. The threshold value is used to extract characteristics from the results of statistical analysis on sample behavior logs. If an inappropriate value is set to the threshold, the system cannot extract appropriate characteristics for accurate detection. The other one is a threshold value used for matching match-target behavior logs with matching templates. The threshold value defines threshold of the degree of conformity between them. If the degree of conformity is more than the threshold value, the logs are regarded as conformable logs to the matching template. If an inappropriate value is set to the threshold, the system cannot successfully detect user behavior to be detected or mistakenly detect user behavior not to be detected.

Because kinds of habitual activities and order of them vary with users, appropriate values depend on user behavior. Therefore, threshold values are preferable to be individually determined for each user. If the values are determined after many individual behavior logs are collected, appropriate values are found by simulating template matching with the collected behavior logs. However, the system can use only a small number of individual sample behavior logs to determine initial threshold values because service providing should be early started after introducing the system into homes. It is difficult to determine threshold values appropriate for individual user only with a small number of sample behavior logs. In addition, it is preferable to utilize both true cases and false cases to determine appropriate values. False cases from which true cases are hardly distinguished are particularly required. Even true cases are small when initial threshold values should be determined. It is impossible to use enough false cases useful for determining appropriate threshold values.

In a basic determination method, developers of a system or experts of the system determine the initial values common to all users before introducing the system into homes without sample behavior logs of individual users, because it is difficult to determine initial threshold values only with a small number of individual sample behavior logs. The determination method uses data of test users to determine the common values. Having a system used by some test users on a trial basis, many sample behavior logs of individual test users are collected. These logs include both true cases and false cases. By simulating template matching with the logs individually, the experts analyze relativity between change of detection accuracy and changes in threshold values. Finally, common threshold values are determined so that detection accuracy is averagely high for test users. The values are used as initial threshold values common to all users after introduction of the system. However, because there are some users whose appropriate threshold values are different from the common threshold values, detection accuracy varies with users. Common threshold values cannot achieve high detection accuracy for some users.

It is necessary to improve detection accuracy of some users whose detection accuracy is low with common threshold values, by determining appropriate initial threshold values.

3.5. Related Works on Setting Threshold

There are several approaches to determine appropriate threshold values in a variety of fields. In image processing, a determination method of a threshold used for extracting a specific area from a target image has been proposed [6]. This method can be used only if both parts to be extracted and parts not to be extracted exist together in a recognition target. The issue of behavior detection does not meet such a condition, because behavior detection in this paper considers whether a match-target behavior log conforms to a matching template or not. This approach in image processing cannot be applied to the issue. In other approaches, Support Vector Machines and boosting have been used for text categorization [3, 15], and Hidden Markov Model is used for speech and gesture recognition [1, 11]. These approaches can determine appropriate threshold values under the assumption that the approaches can collect and analyze many samples of a recognition target or many samples of others which have similar characteristics to samples of the recognition target instead. However, initial threshold values must be determined under the constraint of a small number of sample behavior logs for creating a matching template. In addition, because characteristics of user behavior in homes are different among individual users, behavior logs of other people other than a user cannot be used as sample behavior logs of the user. Although these approaches can be used for learning appropriate threshold values after many personal behavior logs have been collected, these approaches cannot be used for determining initial threshold values appropriate for individuals.

In a field of behavior recognition, most existing studies do not discuss how to determine initial threshold values. They use given values or values which are determined by analyzing many behavior logs.

4. Personalization of Service Activation

4.1. System Overview

Figure 1 shows an overview of the proposed system. This system is composed of RFID-tagged objects, a wearable RFID reader and a home server. The RFID reader reads tag-IDs of objects and sends tag-IDs to the server every time the user touches each object. In the server, the histories of touched objects are stored as behavior logs. After introducing the system into individual home, a certain time period is used for collecting behavior logs of individual user. Services are not provided while that period.

With the collected behavior logs as sample behavior logs, the Matching Template Creator creates matching templates which represent characteristics of user behavior. A matching template is created for every behavior such as leaving home and going to bed. For example, in a case

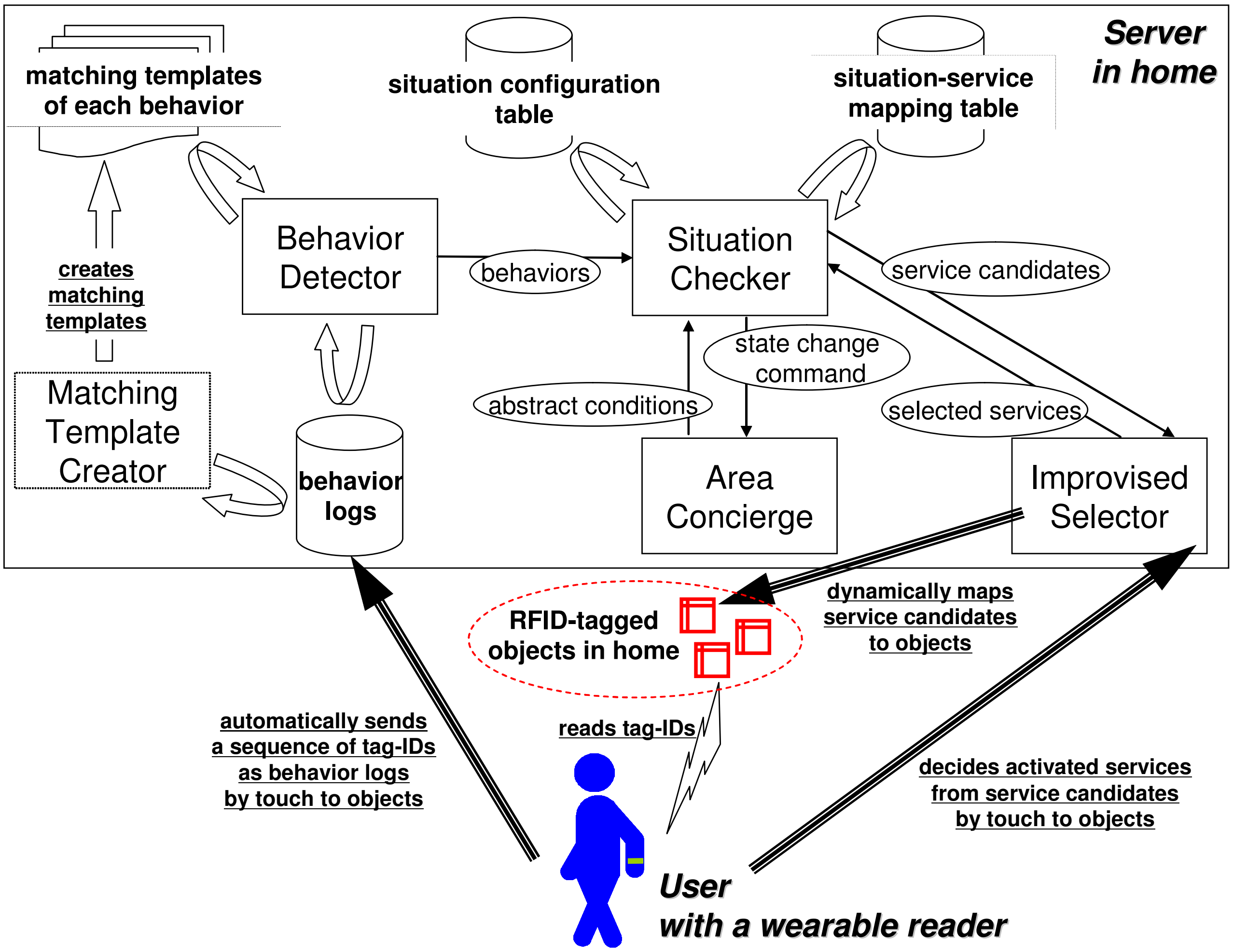

Server in home
matching templates of each behavior
situation configuration table
situation-service mapping table
creates matching templates
Behavior Detector
behaviors
Situation Checker
service candidates
Matching Template Creator
behavior logs
state change command
abstract conditions
selected services
Area Concierge
Improvised Selector
RFID-tagged objects in home
dynamically maps service candidates to objects
automatically sends a sequence of tag-IDs as behavior logs by touch to objects
reads tag-IDs
decides activated services from service candidates by touch to objects
User with a wearable reader

Figure 1. The system with combination of a system-active approach and a user-active approach.

of leaving home, the system shows only behavior logs in which the user touched objects of great relevance to leaving home such as doorknob of the front door and let him select behavior logs of true leaving home. Therefore, the system can exactly use behavior logs of leaving home to create a matching template representing characteristics of leaving home adequately.

After creating matching templates of each behavior, the system starts providing services. First, the Behavior Detector detects behaviors of triggers by matching behavior logs acquired online according to user activities with matching templates every time the user touches something. The Behavior Detector informs the Situation Checker of detected behaviors.

In this system, we refer to a set of conditions to provide services as a *situation*. A situation σ is defined as follows.

$$\sigma = \{b, p, e_1, e_2, ..., e_i, ..., q_1, q_2, ..., q_i, ...\}$$

Here, b is a user behavior to be detected, such as leaving home. p is the user position. e_i denotes the state of an object i which exists in the home, such as "the front door is locked." The variety of the states depends on kinds of sensors combinated to our system. q_i denotes the position of the object i, such as "a wallet is in the bedroom." A variety of situations are defined in the *situation configuration table*.

Referring to the situation configuration table, the Situation Checker checks whether any situations are happening or not. For example, this means even if the system detects the behavior of going to bed when the front door is already locked, a service for warning of unlocked front door is not provided. The states of conditions except user behavior b are always recognized by the Area Concierge which manages and controls sensors and actuators in cooperation with a home-network. The Area Concierge informs the Situation Checker of changes of the states. If there are situations which are happening, the Situation Checker searches services appropriate for the situations by referring

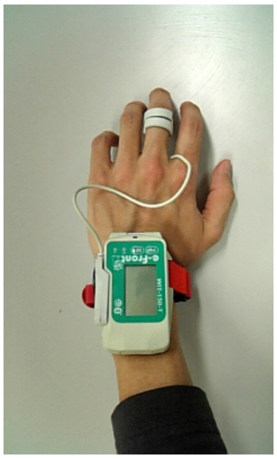


Figure 2. The ring-type RFID reader.

User A : Leave Home	User A : Come Home	User B : Leave Home	User B : Come Home
...	...	...	...
...	...	...	...
toothbrush	key case	toothpaste	bag
lavatory cup	entrance light switch	toothbrush	cell phone
lavatory faucet	pass case	hair dressing	portable music player
wardrobe	cell phone	comb	wallet
hanger	wrist watch	shaver	bicycle key
pants hanger	lavatory faucet	hanger	bag
cell phone	lavatory cup	VCR remote control	hanger
pass case	lavatory faucet	TV switch	lavatory cup
wrist watch	lavatory cup	wallet	lavatory faucet
key case	lavatory faucet	cell phone	lavatory cup
bag	lavatory cup	bicycle key	lavatory faucet
refrigerator	lavatory faucet	portable music player	TV switch
milk carton	hanger	bag	PC mouse
...	...	...	...
...	...	...	...

Figure 3. Examples of behavior log.

to *situation-service mapping table* and informs the Improvised Selector of the services as service candidates.

The Improvised Selector offers the user the service candidates for activating and maps each service candidate to one of objects around him. He chooses and can activate desirable services from the service candidates, just by touching to the objects mapped from the desirable services.

Service candidates are selected by a system-active approach which starts the process for providing services without any particular operation of the user. In addition, combining a user-active approach that the user himself finally decides activated services, our system prevents providing inappropriate services and also achieves service providing along user intention which it is not easy to infer. Of course, depending on provided services, our system can automatically activate the services by omitting the final decision of the user.

4.2. Matching Templates

We have studied how to create matching templates and how to detect user behavior with the templates. In Section 4.2 and Section 4.3, we briefly describe our studying detection method[19, 20]. Figure 3 shows actual behavior logs of two users, which have been recorded using a ring-type RFID reader shown in Figure 2. These are parts of behavior logs of two scenes which are before leaving home and after coming home. In the same scene, kinds of habitual activities and the order of them are different among individual users. In addition, comparing each user's log of leaving home to log of coming home, it is found that a user touches different kinds of objects or touches the same objects in a different order in different situations.

We represent characteristics of a sequence of habitual activities with kinds of activities such as "brushing teeth" and the order of them such as "the user wears his clothes after he brushes his teeth". Because the objects touched by the user significantly indicate kinds of activities and the order of them, our system characterizes user behaviors with kinds of touched objects and the order of touched objects.

With sample behavior logs which are histories of touched objects, our system creates a matching template represented by a set of ordered pairs which show the order relation of contact of a user to objects.

The flow to create a matching template of the user is shown in Figure 4 with an example of a matching template in a scene of leaving home. First, w cases of behavior logs of leaving home are collected as sample behavior logs. The deadline to provide services is the instant the user touches a doorknob of the front door in order to go outside house. We must create a matching template with which the system can detect that the user is leaving home by the instant. Therefore, each sample behavior log is a record of t_l minutes just before the user touches a doorknob of the front door. The time length t_l minutes of a sample behavior log is predetermined. If m objects are sequentially touched in a behavior log l, then l is represented as a conjunction $\{o_1, o_2, \dots, o_i, \dots, o_m\}$, where, $o_{i-1} \neq o_i (1 < i \leq m)$. Second, all ordered pairs between two objects are enumerated from all of collected sample behavior logs. If an object o_j is touched after an object o_i is touched, then an ordered pair p is represented as $\{o_i \rightarrow o_j\}$, which includes a case of $o_i = o_j$. For example, ordered pairs enumerated from a behavior log $\{o_1, o_2, o_3\}$ are $p_1 : \{o_1 \rightarrow o_2\}$, $p_2 : \{o_1 \rightarrow o_3\}$, $p_3 : \{o_2 \rightarrow o_3\}$. Next, the occurrence of each ordered pair is counted up as occurrence count. The occurrence count means not the amount of the number of times that each ordered pair occurred in a sample behavior log, but the number of sample behavior logs including each ordered pair. For example, if an ordered pair occurs in all sample behavior

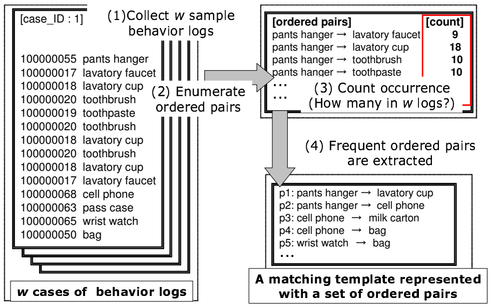

[case_ID : 1]
(1)Collect *w* sample behavior logs
100000055 pants hanger
100000017 lavatory faucet
100000018 lavatory cup
100000020 toothbrush
100000019 toothpaste
100000020 toothbrush
100000018 lavatory cup
100000020 toothbrush
100000018 lavatory cup
100000017 lavatory faucet
100000068 cell phone
100000063 pass case
100000065 wrist watch
100000050 bag
w cases of behavior logs
(2) Enumerate ordered pairs
[ordered pairs] [count]
pants hanger → lavatory faucet 9
pants hanger → lavatory cup 18
pants hanger → toothbrush 10
pants hanger → toothpaste 10
…
…
(3) Count occurrence (How many in *w* logs?)
(4) Frequent ordered pairs are extracted
p1: pants hanger → lavatory cup
p2: pants hanger → cell phone
p3: cell phone → milk carton
p4: cell phone → bag
p5: wrist watch → bag
…
A matching template represented with a set of ordered pairs

Figure 4. How to create a matching template.

logs, the occurrence count of the ordered pair is w. Finally, ordered pairs where ratio of occurrence count to w is more than e are extracted as a matching template Π. e is a threshold for extracting frequent ordered pairs from enumerated ordered pairs. e is referred to as the $extraction\ threshold$.

Two types of ordered pairs composing a matching template are extracted above. One is typified by $\{toothpaste \rightarrow toothbrush\}$. Both a toothpaste and a toothbrush are touched when the user brushes his teeth. This type of an ordered pair represents kinds of habitual activity. The other is typified by $\{toothpaste \rightarrow pants\ hanger\}$. We cannot guess an activity in which the user touches both of a toothpaste and a pants hanger. This type of ordered pair indicates a habitual order of activities, such as the user wears pants after brushing his teeth habitually. Our system can create a matching template which represents characteristics of user behavior by combining two types of ordered pairs.

4.3. Detection of User Behavior

Matching templates are matched with the current behavior log of time length t_l, which is acquired online, every time the user touches objects. Let a set of ordered pairs in the match-target behavior log be Θ. Our system calculates the degree of conformity c of the match-target behavior log to a matching template Π_b for detecting a behavior b with the following formula.

$$c = |\Theta \cap \Pi_b| / |\Pi_b|$$

Here, d is a threshold of the conformity c. If $c \geq d$ then b is detected. d is referred to as the $detection\ threshold$.

In our system, a matching template is composed of ordered pairs which often occurs before the deadline of service providing. Such ordered pairs are composed of objects which are touched with high probability before the deadline. Therefore, because the conformity c significantly increases as the deadline approaches and it becomes more than the threshold d, b is prone to be detected before the deadline.

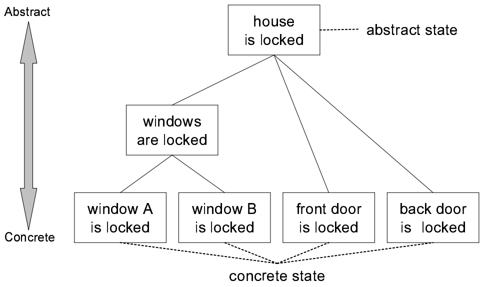


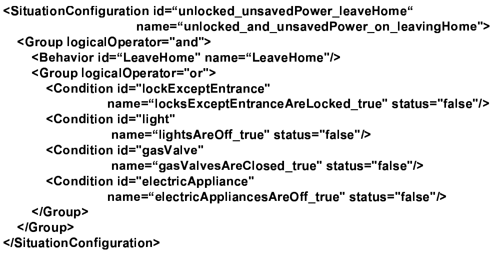
Figure 5. Management of abstract state.

4.4. Situation Check

The Area Concierge defines areas such as the entrance of home, a kitchen and a bedroom in homes hierarchically and manages states of objects, position of the user and position of objects in each area. Also, the Area Concierge can check states about weather, earthquake, and so on by cooperation with outside public sensors. As shown in Figure 5, states are hierarchically managed from concrete level to abstract level. At the most concrete level, each state is individually managed with each sensor, for example, “the window A is locked and the window B is locked”. The Area Concierge also manages at more abstract levels such as “windows are locked”. These hierarchical relations are defined as $concrete\text{-}to\text{-}abstract\ conversion\ rules$. By this way, there is an advantage that it is not necessary to redefine rules at abstract levels when the number of sensors are changed. We only have to redefine part of rules at the most concrete level. This means that our system can be introduced a variety of different homes by customizing only rules at concrete levels along individual user’s home because rules at abstract levels can be defined in advance as rules common to all users. Because it is impossible to represent all states in one hierarchical structure, the Area Concierge manages with some kinds of hierarchical structures. In these hierarchical structures, a state at the most abstract level is referred to as an $abstract\ state$. A state at the most concrete level is referred to as a $concrete\ state$. The Area Concierge informs the Situation Checker of an abstract state and concrete states in the following cases.

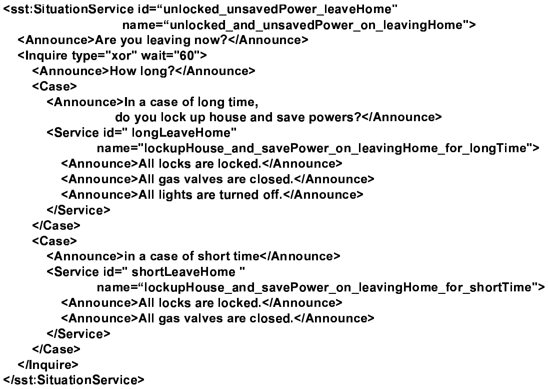
- a case the Area Concierge received an inquiry about states of something from the Situation Checker
- a case an abstract state changed, which is not a case just concrete states changed

In addition, the Area Concierge changes specified states with actuators when received a command to change states of something from the Situation Checker. For example, when received a command “Lock up the house”, the Area



```
<SituationConfiguration id=“unlocked_unsavedPower_leaveHome“
                        name=“unlocked_and_unsavedPower_on_leavingHome">
  <Group logicalOperator="and">
    <Behavior id=“LeaveHome" name=“LeaveHome"/>
    <Group logicalOperator="or">
      <Condition id="lockExceptEntrance"
                 name=“locksExceptEntranceAreLocked_true" status="false"/>
      <Condition id="light"
                 name=“lightsAreOff_true" status="false"/>
      <Condition id="gasValve"
                 name=“gasValvesAreClosed_true" status="false"/>
      <Condition id="electricAppliance"
                 name=“electricAppliancesAreOff_true" status="false"/>
    </Group>
  </Group>
</SituationConfiguration>
```

Figure 6. An example of XML description in situation configuration table.



```
<sst:SituationService id=“unlocked_unsavedPower_leaveHome"
                      name=“unlocked_and_unsavedPower_on_leavingHome">
  <Announce>Are you leaving now?</Announce>
  <Inquire type="xor" wait="60">
    <Announce>How long?</Announce>
    <Case>
      <Announce>In a case of long time,
                 do you lock up house and save powers?</Announce>
      <Service id=" longLeaveHome"
               name="lockupHouse_and_savePower_on_leavingHome_for_longTime">
        <Announce>All locks are locked.</Announce>
        <Announce>All gas valves are closed.</Announce>
        <Announce>All lights are turned off.</Announce>
      </Service>
    </Case>
    <Case>
      <Announce>in a case of short time</Announce>
      <Service id=" shortLeaveHome "
               name=“lockupHouse_and_savePower_on_leavingHome_for_shortTime">
        <Announce>All locks are locked.</Announce>
        <Announce>All gas valves are closed.</Announce>
      </Service>
    </Case>
  </Inquire>
</sst:SituationService>
```

Figure 7. An example of XML description in situation-service mapping table.

Concierge locks windows, the front door and all of others which are related to the command, with referring to the hierarchy from top to bottom.

The position information of the user is acquired by connecting the Area Concierge with medium-range RFID readers and other sensors. It is possible to change kinds of sensors and the number of sensors. The position information of objects shows which area the objects exist in. The position information of objects should be updated in response to move of the objects. However, every object is currently linked with a specific area in advance because this part of our system remains in a development stage.

With behavior detection as a trigger, the Situation Checker refers to the situation configuration table, whose example is shown in Figure 6. The Situation Checker checks whether all conditions of situations composed of the detected behaviors are satisfied or not with the information from the Area Concierge. In the situation in Figure 6, conditions of “lockExceptEntrance”, “light”, “gasValve” and “electricAppliance” are checked. If there are situations whose all conditions are satisfied, the Situation Checker refers to situation-service mapping table for finding services related with the situations. Such services becomes service candidates. Figure 7 shows an example of a situation-service mapping table. In this example, two services can be service candidates.

4.5. Service Activation by Touch-to-Object

The Improvised Selector maps service candidates to objects, which exists in area where the user is, on a one-to-one basis. To decide objects mapped with service candidates, the Improvised Selector makes the user touch objects around him. Because the user decides objects which he can touch easily as switches for choosing services, it is easy for him to understand which objects are mapped with service candidates. Alternatively, the Improvised Selector can automatically decide objects mapped with service candidates if the user wants to omit to decide the objects. Then the Improvised Selector offers contents of service candidates and objects mapped by them. There are some possible ways to offer the user service candidates. Currently, offering is executed by voice announcement. The user chooses desirable services from service candidates by touching to mapped objects. The user’s choice is informed to the Situation Checker and the Situation Checker activates the chosen services. If contents of activated services are related with control of actuators such as lights, the Situation Checker sends commands for controlling actuators to the Area Concierge.

Our system can activate meticulous services along user intention and feeling, which cannot be inferred by the system, by defining multiple services or combination of services, which may be provided when detected one behavior, as service candidates in the situation-service mapping table. Suppose the system detects that a user is leaving his home. At that time, it is not easy to exactly infer his intention, whether he goes away for a long time or for a short time, by computers. For example, our system offers him the following two services.

1. The system locks all windows, turns off all lights, and turns off all air conditioners.

2. The system locks all windows, turns off all lights, but leaves air conditioners as it is.

If he goes away for a long time, he wants to activate the first one. If he goes away for a short time, he wants to activate the second one. In this way, our system provides meticulous services along user intention.

In addition, the system can prevent providing inappropriate services based on false detection. Suppose, before a user goes to bed, he brushes his teeth, goes to the toilet and turns the doorknob of the front door for checking that it is locked. At that time, if the system faultily detects that he is

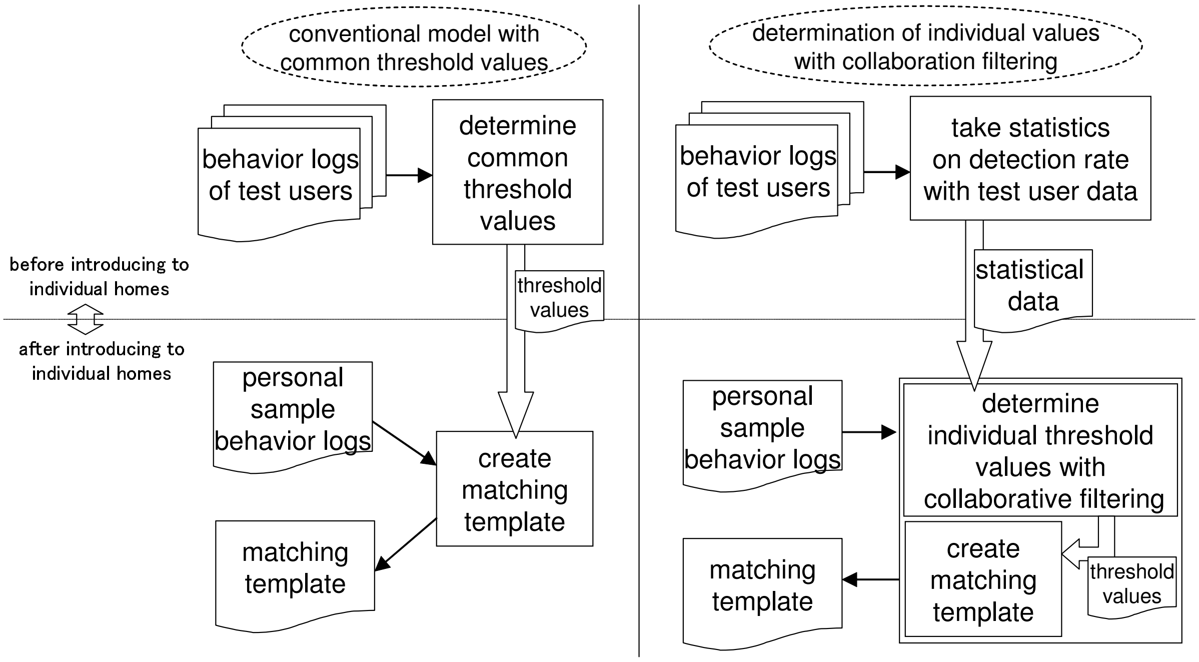

conventional model with common threshold values
behavior logs of test users
determine common threshold values
threshold values
before introducing to individual homes
after introducing to individual homes
personal sample behavior logs
create matching template
matching template
determination of individual values with collaboration filtering
behavior logs of test users
take statistics on detection rate with test user data
statistical data
personal sample behavior logs
determine individual threshold values with collaborative filtering
create matching template
threshold values
matching template

Figure 8. Dynamic determination model with collaborative filtering.

leaving his home and activates a service for turning off an air conditioner, the inappropriate service is not welcome for him. In such a case, our system prevents the inappropriate service if himself finally does not activate the service.

The interface for choosing activated services is expected to be user-friendly so that any users can operate with no stress. In our system, the touch-to-object interface has the following advantages.

- Even users unfamiliar with computers can easily decide whether they activate services or not just by touch-to-object which is an simple and costless operation without being conscious of computers.

- Users can activate services on the spot without moving to specific places because services are dynamically mapped to objects in an area where users are.

5. Personalization of Behavior Detection

5.1. A Model for Utilizing Test User Data

Behavior detection is the most important for context estimation in the proposed system. Our behavior detection method has two parameters whose values should be determined for individual users. They are the extraction threshold and the detection threshold. The proposed system determines initial threshold values dynamically for each user, unlike the conventional model which uses fixed common threshold values. In this paper, a user whose threshold values are determined is referred to as a *target user*. Figure 8 shows the conventional model on the left side, and our model which determines initial threshold values for individuals on the right side. Our model acquires a rule to individually determine the threshold values for each matching template of the target user from the statistics on data of test users. The horizontal center line shows a partition of the two phases for introducing a home context-aware system to the home of the target user. The upper portion is the phase before introducing the system which is referred to as the *development phase*. The lower side is the phase after introducing the system which is referred to as the *operation phase*.

As shown in Figure 8, the conventional model determines common threshold values at the development phase. First, the model collects behavior logs of test users. Next, for every test user, the model repeatedly creates a matching template with the logs, while matching the logs with the matching template based on cross validation. Analyzing the result of detection accuracy on the matching, the model determines the threshold values with which detection accuracy averaged for all test users is the highest. At the operation phase, the model creates an individual matching template with personal behavior logs. However, the threshold values are common irrespective of the target user. In this conventional model, detection accuracy can be low because of differences between common values and the best values of the target user.

To dynamically determine apprppriate threshold values for individuals, it is preferrable to acquire knowledge from personal behavior logs of the target user. However, it is difficult to determine appropriate values only with a small number of personal behavior logs.

Considering the similarity between the target user and each test user, our determination method determines threshold values of the target user, based on the data of test users whose characteristics are similar to characteristics of the target user. To calculate the similarity between the target user and each test user, the method focuses on the average number of ordered pairs composing matching templates of each user. Values of the extraction threshold and the detection threshold are determined by estimating a position of the target user on a feature space, which is composed of behavior detection accuracies and the average number of ordered pairs on every test user, based on an idea of collaborative filtering. As shown in Figure 8, first, our method takes statistics on the average number of ordered pairs of each test user and detection accuracy before introduction of the system. A feature space of this statistical data corresponds to the rule for determinating threshold values. After that, values of two thresholds are determined at the same time by executing collaborative filtering with the statistical data and a small number of personal behavior logs of the target user when a matching template is created after introducing the system into the home of the target user.

5.2. Collaborative Filtering

Collaborative filtering is a process for automatically estimating unknown information of a target user with some known information of him and known information of other users. Here, the informations mean features such as tendency and taste. They have to be able to be collected with a form which can be expressed on the numeric axes. First, the similarity between the target user and each other user is calculated with known information on both the users. Next, unknown information is estimated using known information of other users who are similar to the target user. This estimation is utilized for recommendation or personalization, as used in Amazon.com [7].

Our determination method calculates the similarity between the target user and each test user on a feature of the average number of ordered pairs composing matching templates and estimates behavior detection accuracy of the target user by utilizing statistical data on detection accuracy of test users on every setting of thresholds. The estimation enables to determine threshold values with which detection accuracy is the high.

5.3. Estimation of Initial Threshold Values

With an example of a matching template of leaving home of a target user v, this section describes how to determine threshold values with collaborative filtering. At the development phase, the following steps are executed to calculate detection accuracy of each test user on every setting of two thresholds. Here, w is a given value common to all users.

1. Collect behavior logs of leaving home as true cases and also behavior logs other than leaving home as false cases.

2. Select w true cases as sample behavior logs and create w matching templates with each setting of the extraction threshold value $e = 1/w,\ 2/w,\ ...,\ w/w$, using the w true cases. Here, the number of ordered pairs composing each matching template is recorded.

3. With all settings of the detection threshold d from 0.01 to 1.00, match all true cases and all false cases with the w matching templates.

4. Repeat k times from step 2 to step 3, using new matching templates created with a new combination of w true cases every time.

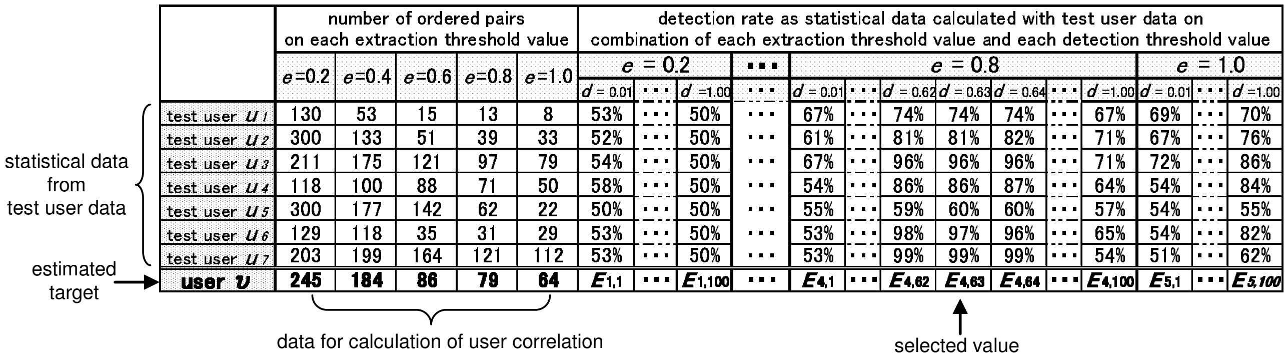
With these steps, $true\text{-}positive\ rate$ (TPR), $true\text{-}negative\ rate$ (TNR), and $half\ total\ true\ rate$ ($HTTR$) are calculated on every setting of thresholds per combination of w true cases by taking statistics on all results of the matchings. The number of threshold settings is $w \times 100$. Here, we explain these three rates with an example of detection of leaving home. TPR means the rate which our detection method successfully detects each subject's leaving home by the deadline, when matching behavior logs of leaving home with matching templates of leaving home. On the other hand, TNR means the rate which the detection method does not detect their leaving home, when matching behavior logs except leaving home with matching templates of leaving home. It is desirable that both TPR and TNR are high. HTTR is an average of TPR and TNR.

After the above steps, the followings are calculated.

- the averate number of ordered pairs composing k matching templates created on each setting of the extraction threshold e

- the average HTTR value on each combination of the setting of the extraction threshold e and the setting of the detection threshold d

These are statistical data which show characteristics of each test user. Next, threshold values of user v are determined when his matching template is created, by collaborative filtering with the statistical data. Figure 9 shows an example



	number of ordered pairs on each extraction threshold value					detection rate as statistical data calculated with test user data on combination of each extraction threshold value and each detection threshold value													
	e=0.2	e=0.4	e=0.6	e=0.8	e=1.0	e = 0.2			▪▪▪	e = 0.8							e = 1.0		
						d = 0.01	▪▪▪	d =1.00	▪▪▪	d = 0.01	▪▪▪	d = 0.62	d = 0.63	d = 0.64	▪▪▪	d =1.00	d = 0.01	▪▪▪	d =1.00
test user u_1	130	53	15	13	8	53%	▪▪▪	50%	▪▪▪	67%	▪▪▪	74%	74%	74%	▪▪▪	67%	69%	▪▪▪	70%
test user u_2	300	133	51	39	33	52%	▪▪▪	50%	▪▪▪	61%	▪▪▪	81%	81%	82%	▪▪▪	71%	67%	▪▪▪	76%
test user u_3	211	175	121	97	79	54%	▪▪▪	50%	▪▪▪	67%	▪▪▪	96%	96%	96%	▪▪▪	71%	72%	▪▪▪	86%
test user u_4	118	100	88	71	50	58%	▪▪▪	50%	▪▪▪	54%	▪▪▪	86%	86%	87%	▪▪▪	64%	54%	▪▪▪	84%
test user u_5	300	177	142	62	22	50%	▪▪▪	50%	▪▪▪	55%	▪▪▪	59%	60%	60%	▪▪▪	57%	54%	▪▪▪	55%
test user u_6	129	118	35	31	29	53%	▪▪▪	50%	▪▪▪	53%	▪▪▪	98%	97%	96%	▪▪▪	65%	54%	▪▪▪	82%
test user u_7	203	199	164	121	112	53%	▪▪▪	50%	▪▪▪	53%	▪▪▪	99%	99%	99%	▪▪▪	54%	51%	▪▪▪	62%
user v	**245**	**184**	**86**	**79**	**64**	$E_{1,1}$	▪▪▪	$E_{1,100}$	▪▪▪	$E_{4,1}$	▪▪▪	$E_{4,62}$	$E_{4,63}$	$E_{4,64}$	▪▪▪	$E_{4,100}$	$E_{5,1}$	▪▪▪	$E_{5,100}$

Figure 9. Estimation of values for a target user by collaborative filtering with test user data.

of the statistical data and the data of user v, which are used for collaborative filtering. In the example, w is 5. Rows from "test user u_1" to "test user u_7" are the statistical data from above calculation.

Information of how many ordered pairs compose each matching template on each setting of e and information how high detection accuracy is brought with each combination of the setting of two thresholds are obtained from the statistical data. On the other hand, there is only information, obtained from a small number of personal sample behavior logs, of user v when his initial matching template is created. As shown in the bottom row of Figure 9, the determination method utilizes the number of ordered pairs composing each matching template which is created on each setting of e with personal sample behavior logs of user v. At this time, it is unknown how high detection accuracy is brought with each matching template of user v. By collaborative filtering, first, user correlation between user v and each test user is calculated with 5 values of the number of ordered pairs, which are known information of both user v and each test user. The user correlation shows the similarity between users. Second, HTTR values of matching template of user v is estimated with HTTR values of rows from test user u_1 to test user u_7, based on the calculated user correlations. From $E_{1,1}$ to $E_{5,100}$ show the estimated HTTR values. Here, $E_{i,j}$ means the estimated HTTR value on the setting where e = i/w and d = $j/100$. After the estimation, the determination method selects one estimated value from all estimated values as follows.

1. Select the maximum estimated value.

2. If more than two estimated values are selected in the above step, pick up the longest sequence of the maximum estimated values and select the estimated value in the center of the sequence.

3. If more than two estimated values are selected in the above step, select an estimated value on the smallest value of e from the remaining candidates.

Suppose three rows of $\{E_{4,62}, E_{4,63}, E_{4,64}\}$ are the longest sequence of the maximum estimated values in Figure 9. In such a case, $E_{4,63}$ in the center of three values is selected. Finally, because $E_{4,63}$ is the estimated value on e = 0.8 and d = 0.63, these values are determined as threshold values for user v.

6. An Experimental Life Space

6.1. Implementation

We have built an experimental life space by implementing the proposed system. Figure 10 shows the life space. The life space is Japanese style house, where users take off their shoes when they go into the house. The life space is composed of some areas such as entrance, living, kitchen, and dining. Real furniture and electric appliances are equipped. About 1000 passive RFID tags whose frequency is 13.56 MHz are installed in 159 objects in all areas. Figure 11 shows examples of the objects. Behavior logs of users who wear RFID readers on their hand can be collected in this life space. In addition, middle-range RFID readers are installed into this life space as devices for acquiring position information of users.

6.2. Data Collection for Experiments

We have collected behavior logs in the experimental life space to conduct experiments for evaluation of the proposed system, with 8 experimental subjects "A" to "H". Target behaviors to be detected are 4 behaviors which are leaving home, coming home, getting up and going to bed. Previously we had a questionnaire with 17 men and 4 women to decide the target behaviors. As a result, above 4 behaviors

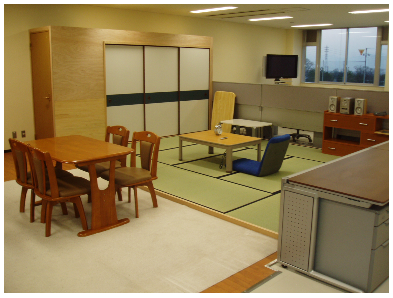


Figure 10. An experimental life space.

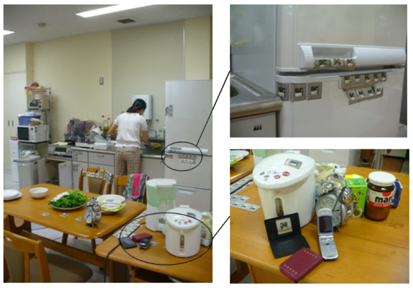


Figure 11. Objects installed with RFID tags.

have been selected because user's mistakes are effectively prevented by services triggered by detection of these 4 behaviors.

Prior to collect behavior logs as data for the experiment, we have had a survey questionnaire for 2 weeks to confirm that users have habitual characteristics of 4 behaviors. In the questionnaire, subjects have described complete details of kind of objects they touched and the order of touched objects during 10 minutes before leaving home, after coming home, after getting up and before going to bed. We have confirmed the following things from their description.

- Some activities are interleaved in a 10 minutes sequence of activities.
- Subjects do not finish a sequence of activities within 10 minutes after coming home and after getting up.

Although there are no definite deadlines of providing services after coming home and after getting up, we must decide the deadlines to clear up success or failure of behavior detection in experiments. In view of the above confirmation, we have set the deadline to "10 minutes after subjects open the front door and enter the house" when coming home. Similarly, we have set the deadline to "10 minutes after subjects get out of bed" when getting up.

As experimental data, we have collected behavior logs of 4 behaviors in a database of a server by sensing actual objects which 8 subjects have touched in the experimental life space shown in Figure 10 and Figure 11. We have had subjects touch the objects with being aware of the position of tags on the objects. However, we have not forced subjects to keep touching the objects so that tag-IDs are exactly read out. Therefore, though the reading accuracy of the reader is not bad, part of touched objects may be not recorded as behavior logs.

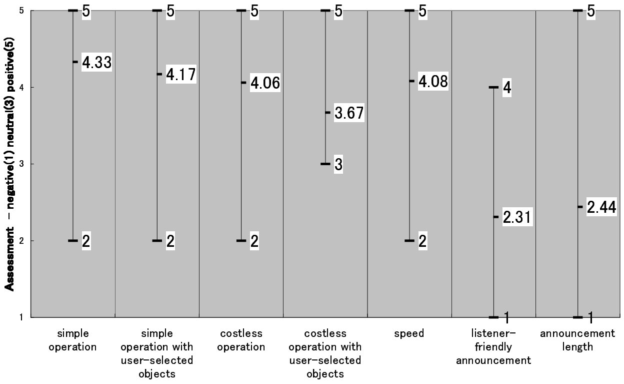


Figure 12. 5-point assessment of the touch-to-object interface.

7. Usability of the Touch-to-Object Interface

7.1. Experiment for the Interface

We have conducted an experiment to evaluate the usability of the touch-to-object interface for deciding services to be activated, which is indispensable to actualize service activation by users in a user-active approach of our system. In the experiment, we ask subjects to decide activated services from service candidates by touching actual objects in the above experimental life space before leaving home and before going to bed. The service candidates are offered with voice announcement. The situation configuration table and the situation-service mapping table have been predefined. Each subject experiences both a case where the system automatically decides objects mapped with service candidates and a case he decides the objects by touching some objects around him prior to decision of services to be activated. Subjects answer a questionnaire after above experiences.

7.2. Discussion on Usability of the Interface

Subjects have answered their opinions with free description in the questionnaire. In addition, they have evaluated the usability of the interface with 5 respects which are simple operation for activating services, costless operation for activating services, speed of service activation, listner-friendliness of announcement and lenghth of announcement, based on a 5-point Likert scale[4]. With points between 1 to 5, subjects answer their evaluation on a statement such as “they can be satisfied with the simplicity of operation”. The points of answer mean that 1:Strongly disagree, 2:Disagree, 3:Neither agree nor disagree, 4:Agree and 5:Strongly agree. Figure 12 shows the average points of all subjects. On the simple operation and the costless operation, the points of “with user-selected objects” are also shown. The points mean evaluation only in a case subjects decide the objects mapped with service candidates.

As a result, the average points are more than 4 in respect of simple operation, costless operation and speed. Subjects highly valued the touch-to-object interface because they can choose services by simply touching objects without being conscious of computers. Comparing a case where the system decides objects which become switches with a case where subject decides the objects, the average points are lower in the latter case. As a cause of this difference, there is an opinion that it is annoying to touch objects more than once to decide switches and to choose desirable services. However, on the other hand, there are the following opinions from most of subjects.

- If the system automatically decides the objects mapped with service candidates, users sometimes cannot intuitively image the relation between services and objects.
- It is more easy-to-understand to decide the objects by users themselves than the objects are automatically decided.

These opinions indicate that the touch-to-object interface has both advantages and disadvantages. But advantages are more significant for most users, because users who feel annoyed with touching to objects several times can make our system choose the objects automatically. In addition, most subjects valued the following advantage.

- Users can activate services on the spot without moving, because service candidates are dynamically mapped to objects around them.
- Users can comfortably accept services from the system, because they can choose activated services with a simple interface without leaving service choice to computers.

Many opinions from subjects prove the touch-to-object interface has high operability to actualize service activation by users.

Although the touch-to-object interface has received high evaluation, the average points are less than 3 in respect of listener-friendliness of announcement and length of announcement. There is a problem that if all of detailed contents which should be reported to users are announced the length of voice announcement tends to become long. It can be stress for users. Also, there is an opinion that if users do not listen to the voice announcement carefully they may miss the relation between services and objects because the voice announcement is invisible and intangible.

In the future, we will study visualization of relation between services and objects by combinating voice announcement with displaying on an information terminal to resolve annoyance on the dynamic mapping. Display of the relation is expected to make users understand the relation clearly. In addition, it may be effective to report short abstract contents with voice announcement and to display concrete contents for reduction of user stress.

8. Accuracy of Behavior Detection

8.1. How to Calculate Detection Rates

We have conducted experiments to evaluate our method for behavior detection with the collected data described in section 6. In the experiment, detection accuracies of 4 behaviors, which are leaving home, coming home, getting up, and going to bed, are calculated with behavior logs. Detection accuracies are calculated both with threshold values common to all users and with threshold values determined for individuals. The latter results are compared with the former results.

To calculate the accuracies, matching templates are created with part of collected behavior logs and each matching template is repeatedly matched with behavior logs which are not used for creating the template. In the experiments, the ground truth is given by subjects themselves.

Compared with 4 behaviors of our target, subjects touch entirely-different kinds of objects in scenes such as reading books, cooking and having a meal. Our detection method can easily distinguish between behaviors in these scenes and the 4 behaviors. To calculate the accuracies in the experiment, behavior logs which are prone to be faultily detected are adequate as false cases. Therefore, we use behavior logs of a behavior as true cases from target 4 behaviors and those of other 3 behaviors as false cases.

To verify that our detection method detects user behavior by the deadline, we set the time length t_l of sample behavior logs and match-target behavior logs to 10 minutes in experiments. Behavior logs of leaving home are logs of past 10 minutes to the time subjects touch the doorknob of the front door. Behavior logs of going to bed are logs of past 10 minutes to the time subjects lie down on the bed for sleeping. Behavior logs of coming home are logs of 10 minutes from the time subjects touch the doorknob of the front door. Behavior logs of getting up are logs of 10 minutes from the time subjects get out of bed. That is, if the conformity c which is calculated by matching true cases with matching templates is more than the detection threshold d, it means right behaviors are detected by the deadline.

TPR and TNR are calculated as follows. First, statistical data of 8 subjects for collaborative filtering are calculated with their behavior logs, based on the method described in the previous section. Next, the following steps are executed on each subject to calculate TPR and TNR with threshold values determined for individuals. In experiments, each of subjects is considered as a target user and other subjects are considered as test users.

1. Select 5 true cases and create a matching template with the cases, based on the extraction threshold e.
2. Select other 1 true case and match the case with the matching template.
3. Match all of false cases with the matching template, with the detection threshold d.
4. Repeat 100 times from step 1 to step 3, using a new matching template created with a new combination of 5 true cases every time.

Here, TPR is calculated based on cross validation. TNR is calculated by matching all false cases with all created matching templates. The number of sample behavior logs for creating a matching template is set to 5, which can be collected within a week. This is because our study assumes that our system must start providing services to users within a week at the latest since the beginning of use of our system. The values of e and d are determined when each matching template is created in step 1 by collaborative filtering using statistical data of 7 subjects other than the target user whose TPR and TNR are calculated in the above steps. In experiments, if the number of ordered pairs are more than 300, it is calculated as 300 because more than 300 ordered pairs are empirically too many as the number of characteristics of user behavior. Because values of statistical data used for collaborative filtering must be normalized, all values are normalized so that the values are between 0 to 300. From the result of all matchings, TPR, TNR, and HTTR of every subject are calculated on the case with threshold values determined for individuals.

After that, these rates with common threshold values are calculated by similar steps. In this case, e is fixed to 0.8. d are 0.33 in leaving home, 0.31 in coming home, 0.47 in getting up and 0.63 in going to bed. These values have been determined in advance so that detection accuracies are the highest.

8.2. Detection with Common Threshold Values

First, this section shows detection accuracies with common threshold values. We have previously reported experiments on behavior detection with common threshold values[18, 19]. Table 1 shows a result of comparing detection accuracies of our detection method using ordered pairs and detection accuracies of a detection method with matching templates represented as Hidden Markov Model (HMM) which is often used for behavior recognition analyzing time-series patterns by existing methods. TPR and TNR on Equal Error Rate (EER) at which difference between TPR and TNR is the smallest are respectively shown in the table. Each rate is an average rate of all subjects. The accuracies with ordered pairs are higher than those with HMM. There are differences more than 10% about leaving home. Moreover, those are more than 30% about other behaviors. The output probability of HMM falls remarkably in a case that rare activities are inserted into an observed sequence of user activities. It falls also in a case that users change the order of part of activities in the sequence. The differences between accuracies with ordered pairs and accuracies with HMM are proof of that ordered pairs are robuster to such complex user activities than HMM.

Table 1. Detection rate on EER.

	ordered pairs		HMM	
behavior	TPR(%)	TNR(%)	TPR(%)	TNR(%)
leave home	95.25	92.94	79.25	79.17
come home	92.38	95.91	62.00	60.73
get up	85.00	80.45	53.00	56.46
go to bed	80.50	83.50	46.13	46.12

However, detection accuracies of some users are not enough high. Differences between common threshold values and values appropriate for each user affect on the detection accuracies. 4 tables, which are from Table 2 to Table 5, show differences between the common value of the detection threshold and the best value of the detection threshold for each subject. The best values are calculated by analyzing the results. The common value for each behavior is shown in the bottom row of the tables. In addition, TPR and TNR with the common threshold values are shown together in the tables. Differences between common values and the best values of each subject are not relatively big on leaving home and coming home. On the other hand, there are more differences of those values on getting up and going to bed. Accordingly, detection accuracies on getting up and going to bed are overall less than detection accuracies on leaving home and coming home. In addition, there are differences among the best values of subjects. Comparing detection accuracies with differences between common values and the best values of each subject in each table, it is apparent that the more differences bring lower detection accuracies. Detection accuracies of subjects A, G, H in Table 4 and subjects E, H in Table 5 indicate such trend significantly. The detection threshold value directly affects detection accuracy of user behavior. These results show that it is important to

Table 2. Variation of the best value of detection threshold on “leaving home”.

subject	TPR (%)	TNR (%)	best value	difference
A	94.00	96.02	28%	5
B	98.00	85.44	40%	7
C	78.00	83.20	46%	13
D	95.00	98.00	23%	10
E	99.00	98.96	34%	1
F	96.00	97.00	28%	5
G	100.00	96.36	35%	2
H	98.00	95.18	36%	3
common threshold value			33%	-

Table 3. Variation of the best value of detection threshold on “coming home”.

subject	TPR (%)	TNR (%)	best value	difference
A	89.00	95.93	31%	0
B	99.00	98.12	35%	4
C	81.00	83.37	42%	11
D	98.00	78.40	56%	25
E	93.00	99.60	24%	7
F	99.00	100.00	30%	1
G	100.00	96.80	50%	19
H	100.00	98.27	43%	12
common threshold value			31%	-

Table 4. Variation of the best value of detection threshold on “getting up”.

subject	TPR (%)	TNR (%)	best value	difference
A	73.00	99.12	20%	27
B	90.00	96.78	47%	0
C	63.00	84.35	43%	4
D	100.00	99.22	55%	8
E	64.00	87.32	45%	2
F	97.00	99.68	46%	1
G	100.00	74.33	67%	20
H	56.00	83.60	28%	19
common threshold value			47%	-

Table 5. Variation of the best value of detection threshold on “going to bed”.

subject	TPR (%)	TNR (%)	best value	difference
A	62.00	85.34	45%	18
B	91.00	71.84	64%	1
C	95.00	96.92	72%	9
D	78.00	94.66	68%	3
E	28.00	91.24	40%	23
F	95.00	99.14	47%	16
G	98.00	99.32	61%	2
H	58.00	100.00	36%	27
common threshold value			63%	-

determine threshold values for individuals.

8.3. Detection with Individual Threshold Values

As results of experiments, detection accuracies with threshold values determined for individuals are shown in 4 tables, which are from Table 6 to Table 9. The tables respectively show the results of leaving home, coming home, getting up, and going to bed.

Behavior detection method must achieve high accuracy stably for behaviors of many users. It is preferable that accuracies of all users are reasonably high rather than that accuracy are very high only for some users and are low for others. As results of experiments, there are some subjects whose TPR or TNR are lower with individual threshold values than those of common threshold values. However, detection accuracies are still high on most of them. They are more than 80%. On the other hand, there are cases where the individual threshold values achieve higher accuracies on some subjects whose accuracies are originally high with common threshold values. Here, these results are not focused on. The following characteristic differences between accuracies with individual threshold values and accuracies with common threshold values are focused on.

- Individual threshold values achieve higher accuracies than low accuracies which are less than 80% with common threshold values.
- Individual threshold values bring lower accuracies, which are less than 80%, than accuracies with common threshold values.

Based on the result of the t-test, the experimental results are evaluated with the idea that difference of more than 5% is a statistically-significant difference between accuracies with individual threshold values and accuracies with common threshold values.

In tables, the differences are shown in parenthesis of each value of TPR and TNR, except the differences which are less than a statistically-significant difference. Positive values mean that individual threshold values have increased detection accuracies. TPR and TNR with the common threshold values are shown in Table 2, Table 3, Table 4 and

Table 6. Detection accuracy of "leaving home" with threshold values estimated by collaborative filtering.

note	subj.	TPR (%)	TNR (%)
	A	95.00	94.46
	B	98.00	82.68
#3	C	96.00 (+18)	57.28 (-25.92)
	D	94.00	91.54
	E	99.00	85.92
	F	90.00	97.44
	G	100.00	95.92
	H	86.00	92.68

Table 7. Detection accuracy of "coming home" with threshold values estimated by collaborative filtering.

note	subj.	TPR (%)	TNR (%)
	A	90.00	97.42
	B	98.00	99.82
	C	79.00	92.60
#1	D	98.00	85.63 (+7.23)
	E	96.00	99.42
	F	98.00	100.00
	G	100.00	92.72
	H	100.00	98.18

Table 8. Detection accuracy of "getting up" with threshold values estimated by collaborative filtering.

note	subj.	TPR (%)	TNR (%)
#1	A	81.00 (+8)	98.07
	B	90.00	92.58
#1	C	78.00 (+15)	82.20
	D	100.00	96.83
	E	62.00	81.23
	F	98.00	99.42
#2	G	100.00	65.72 (-8.62)
#3	H	72.00 (+16)	69.72 (-13.88)

Table 9. Detection accuracy of "going to bed" with threshold values estimated by collaborative filtering.

note	subj.	TPR (%)	TNR (%)
#1	A	69.00 (+7)	80.16
#2	B	95.00	63.94 (-7.9)
	C	96.00	92.48
	D	79.00	84.04
#1	E	48.00 (+20)	84.02
	F	99.00	97.74
	G	100.00	97.62
#1	H	70.00 (+12)	99.10

Table 5. Results are categorized into three groups, #1, #2 and #3. Individual threshold values have achieved higher accuracies in 6 cases of #1. In particular, on TPRs of subject E and H in Table 9, the individual values have improved them +20 and +12 respectively. While, accuracies are lower than that with common threshold values only in 2 cases of #2, which are subject G in Table 8 and subject B in Table 9. In 2 cases of #3, TPR is higher but TNR is lower with individual threshold values. Our determination method is not necessarily effective on these cases. These experimental results show our method can improve detection accuracies of users whose detection accuracies are low with common threshold values, by setting appropriate values to thresholds for individuals.

Because of a property of collaborative filtering, if there is no test user who has strong correlation with the target user at all then the values of the target user are not accurately estimated. It can be a cause of ineffectiveness of our determination method in a few cases.

An additional experiment, which uses all of 8 subjects including an estimated target subject as test users for collaborative filtering, has been conducted. It means that test users include a test user who has likely strong correlation with the target user. The experimental results are shown in Table 10, Table 11, Table 12, and Table 13. As a result, compared to the former experiment, improvement of detection accuracy has been shown in 3 cases. First, TPR of subject C in Table 10 has been improved without decreasing TNR. The difference on TNR of subject G in Table 12 changes from -8.62 to +5.62. In addition, TPR of subject H in Table 13 is improved from +12 to +23. These results indicate that our determination method have a possibility to improve detection accuracy of more users. With these results in mind, to solve above problems, it is important to prepare test users who has strong correlation with the target user by increasing the number of test users and the diversity of test users. To make diverseness of test users, a method for making additional test users artificially is necessary.

9. Challenges for Improvement

Our system can be improved more in the future as follows to personalize the system depending on each user.

The situation configuration table, the service-object

Table 10. Detection accuracy of "leaving home" with threshold values estimated by collaborative filtering which includes a target user in test users.

note	subj.	TPR (%)	TNR (%)
	A	97.00	95.30
	B	98.00	81.84
#1	C	88.00 (+10)	87.68
	D	95.00	98.00
	E	99.00	99.18
	F	95.00	95.76
	G	100.00	95.82
	H	99.00	90.80

Table 11. Detection accuracy of "coming home" with threshold values estimated by collaborative filtering which includes a target user in test users.

note	subj.	TPR (%)	TNR (%)
	A	90.00	97.40
	B	98.00	99.82
	C	80.00	89.72
#1	D	98.00	89.72 (+11.32)
	E	96.00	99.42
	F	98.00	100.00
	G	100.00	97.42
	H	100.00	98.18

Table 12. Detection accuracy of "getting up" with threshold values estimated by collaborative filtering which includes a target user in test users.

note	subj.	TPR (%)	TNR (%)
#1	A	83.00 (+10)	98.83
	B	89.00	94.55
#3	C	79.00 (+16)	79.33 (-5.02)
	D	100.00	97.35
	E	64.00	82.77
	F	99.00	99.90
#1	G	100.00	79.95 (+5.62)
#3	H	74.00 (+18)	65.65 (-17.95)

Table 13. Detection accuracy of "going to bed" with threshold values estimated by collaborative filtering which includes a target user in test users.

note	subj.	TPR (%)	TNR (%)
#1	A	69.00 (+7)	80.16
#2	B	95.00	64.02 (-7.82)
	C	96.00	92.48
#1	D	91.00 (+13)	94.02
#1	E	48.00 (+20)	84.02
	F	99.00	97.74
	G	100.00	97.62
#1	H	81.00 (+23)	100.00

mapping table and the concrete-to-abstract conversion rules are predefined in this system. General contents common to all users are described in these. However, they are not always appropriate for all users. The rules can be redefined, but it is not easy for users unfamiliar with computers to customize these by themselves at present. We must build a method for customizing these easily without complex operation.

This system uses passive RFID system to develop with as few types of sensors and low-cost sensors as possible at present. Later, we will study the possibility of a variety of applications by adding other sensors to this system. For example, we need to add reasonable sensors which achieve acquiring the position information of users and objects more precisely. Checking usefulness and cost of a variety of sensors, we will consider personalization of configuration of sensors combined with our system according to user's budget.

Currently, we have implemented a behavior detection which is appropriate for an application to prevent mistakes and dangers of users. If we implement other behavior detection for different applications in the future, we can extend our system by adding new modules into Matching Template Creator and Behavior Detector. The extentensibility enables our system to personalize kinds of applications depending on each user.

10. Conclusion

In this paper, we have proposed a home context-aware system which has a mechanism for personalization of service activation and personalization of context estimation. The system provides services by combining a system-active approach and a user-active approach. Because users themselves finally choose activated services with a touch-to-object interface in a user-active approach, the system can activate a variety of services also along user intention which cannot be inferred with computers. In a system-active approach, user behavior is detected as a trigger of service providing. The system determine threshold values appropriate for each user in the detection method by utilizing statistical data of test users whose characteristics are similar to each user. The determination enables stable behavior detection. In experiments, we have demonstrated the high possibility of the proposed system.

Acknowledgments

This work was partially supported by Grant-in-Aid for Scientific Research in Japan.

References

[1] T. Asami, K. Iwano, and S. Furui. A stream-weight and threshold estimation method using adaboost for multi-stream speaker verification. In *Proc. the 2006 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2006)*, volume 5, pages 1081–1084, 2006.

[2] J. Barbič, A. Safonova, J.-Y. Pan, C. Faloutsos, J. K. Hodgins, and N. S. Pollard. Segmenting motion capture data into distinct behaviors. In *Proc. the 2004 conference on Graphics interface*, pages 185–194, 2004.

[3] L. Cai and T. Hofmann. Text categorization by boosting automatically extracted concepts. In *Proc. the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval (SIGIR2003)*, pages 182–189, 2003.

[4] T. K. Harris and R. Rosenfeld. A universal speech interface for appliances. In *Proc. the 7th International Conference on Spoken Language Processing (ICSLP 2004)*, pages 249–252, 2004.

[5] T. Huỳnh, U. Blanke, and B. Schiele. Scalable recognition of daily activities with wearable sensors. In *Proc. the 3rd International Symposium on Location- and Context-Awareness (LoCA2007), LNCS 4718*, pages 50–67, 2007.

[6] Y. Kimura, D. Watabe, H. Sai, and O. Nakamura. New threshold setting method for the extraction of facial areas and the recognition of facial expressions. In *Proc. the IEEE Electrical and Computer Engineering, Canadian Conference (CCECE/CCGEI2006)*, pages 1984–1987, 2006.

[7] G. Linden, B. Smith, and J. York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, 2003.

[8] B. Logan, J. Healey, M. Philipose, E. M. Tapia, and S. Intille. A long-term evaluation of sensing modalities for activity recognition. In *Proc. the 9th International Conference on Ubiquitous Computing (UbiComp2007), LNCS 4717*, pages 483–500, 2007.

[9] D. J. Moore, I. A. Essa, and M. H. HayesIII. Exploiting human actions and object context for recognition tasks. In *Proc. the IEEE International Conference on Computer Vision 1999 (ICCV'99)*, pages 80–86, 1999.

[10] J. Nichols and B. A. Myers. Controlling home and office appliances with smart phones. *IEEE Pervasive Computing*, 5(3):60–67, 2006.

[11] F. Niu and M. Abdel-Mottaleb. HMM-based segmentation and recognition of human activities from video sequences. In *Proc. the 2005 IEEE International Conference on Multimedia and Expo (ICME2005)*, pages 804–807, 2005.

[12] D. J. Patterson, D. Fox, H. A. Kautz, and M. Philipose. Fine-grained activity recognition by aggregating abstract object usage. In *Proc. the 9th IEEE International Symposium on Wearable Computers (ISWC2005)*, pages 44–51, 2005.

[13] M. Perkowitz, M. Philipose, D. J. Patterson, and K. Fishkin. Mining models of human activities from the web. In *Proc. the 13th International World Wide Web Conference (WWW 2004)*, pages 573–582, 2004.

[14] J. Riekki, T. Salminen, and I. Alakärppä. Requesting pervasive services by touching RFID tags. *IEEE Pervasive Computing*, 5(2):40–46, 2006.

[15] J. G. Shanahan and N. Roma. Boosting support vector machines for text classification through parameter-free threshold relaxation. In *Proc. the 12th International Conference on Information and knowledge Management (CIKM2003)*, pages 247–254, 2003.

[16] K. Tsukada and M. Yasumura. UbiFinger: Gesture input device for mobile use. In *Proc. the 5th Asia Pacific Computer Human Interaction (APCHI2002)*, pages 388–400, 2002.

[17] S. Wang, W. Pentney, A.-M. Popescu, T. Choudhury, and M. Philipose. Common sense based joint training of human activity recognizers. In *Proc. the 20th International Joint Conference on Artificial Intelligence (IJCAI2007)*, pages 2237–2242, 2007.

[18] H. Yamahara, T. Soma, F. Harada, H. Takada, Y. Shimada, and H. Shimakawa. Tagged World: an intelligent space providing services by interaction between a user and an environment. In *Proc. the 2nd International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM2008)*, pages 333–342, 2008.

[19] H. Yamahara, H. Takada, and H. Shimakawa. Detection of user mode shift in home. In *Proc. the 4th International Symposium on Ubiquitous Computing Systems (UCS2007), LNCS 4836*, pages 166–181, 2007.

[20] H. Yamahara, H. Takada, and H. Shimakawa. An individual behavioral pattern to provide ubiquitous service in intelligent space. *WSEAS TRANSACTIONS on SYSTEMS*, 6(3):562–569, 2007.

Dynamic Service Synthesis on a Large Service Models of a Federated Governmental Information System

Riina Maigre[1], Peep Küngas[2], Mihhail Matskin[3,4], Enn Tyugu[1]
[1]Institute of Cybernetics at Tallinn University of Technology, Tallinn, Estonia
[2]SOA Trader, Ltd, Tallinn, Estonia
[3]Royal Institute of Technology – KTH, Stockholm, Sweden
[4]Norwegian University of Science and Technology – NTNU, Trondheim, Norway
riina@cs.ioc.ee, peep@soatrader.com, misha@imit.kth.se, tyugu@cs.ioc.ee

Abstract

In this paper we describe our experiments with large syntactic Web service models of a federated governmental information system for automatic composition of services. The paper describes a method for handling syntactic service models for synthesis of compound services. The method's implementation as a visual tool developed in software environment CoCoViLa is explained on an example from e-government domain. Given a specification and a goal, the tool automatically synthesizes a program that generates a required composite service description in BPEL or OWL-S.

Keywords: automatic service composition; large service models; e-government services.

1. Introduction

The present paper describes experiences of composition of Web services on very large syntactic Web service models, and it is an extended version of the paper [1] at the 3rd International Conference on Internet and Web Applications and Services. The lessons learned presented here are based on a work with software developed for providing services to citizens by a number of governmental agencies. A federated e-government information system [2], complying to service-oriented architecture, has been developed in Estonia during the recent years. A syntactic service model of a part of the system exists and can be used for automating composition of new services. However, this process is still too complicated for end users and can only be useful, first of all, for software experts developing and maintaining the system.

Developing the federated information system has been a complicated task that has required cooperation of a number of government agencies that already provide services for citizens. Analysis of the system has resulted in a unified service model which includes about three hundred of atomic services [3], including a number of rather primitive data transformers needed for interoperability of databases. Totally more than a thousand atomic services are available, which could be included in the service model and composed into complex services. To determine a structure of a new complex service we are going to use a syntactic service model that describes only inputs and outputs of atomic services and includes references to the semantics of these services.

Although there does not seem to be any literature available considering usage of automated composition tools for federated governmental information systems, many EU countries have started their public sector semantic interoperability initiatives.

The primary aim of this work is development of a tool for automatic composition of Web services that involve atomic services from several governmental institutions. It is easy for the end user to get a service from a governmental agency through the agency's portal. Operations get more complicated when one needs to use services from several agencies, i.e., from multiple providers. Currently an end-user has to get information from one provider and forward the results manually to the second provider's service. The user has to know exactly which data has to be passed from one portal to another.

Manual construction of a new complex service from atomic services is a challenging task even for the software developers, because service descriptions from different providers are published on different servers and the number of possible inputs, outputs and their combinations is large. Our tool is intended to automate this process for software developers. To achieve our goal we use a visual programming environment CoCoViLa [4] that uses automatic synthesis of algorithms and can generate Java code from both visual and textual specifications. The syntactic service model is presented as a specification to the tool developed in CoCoViLa. For each requested service, a goal is given that specifies the input and output data of the service. From this information, an algorithm of the service is composed, and a service description is generated in BPEL (or in OWL-S, if requested).

The paper is structured as follows. The process of service composition is described in Section 4 and Section 5 after discussing the federated e-government information system and its service model in Section 2 and Section 3. Related work is presented in Section 6 and concluding remarks in Section 7.

2. X-road

The central part of Estonian e-government information system is the infrastructure, called X-Road, guaranteeing secure access to nearly all Estonian national databases over the Internet [2]. It is the environment through which hundreds of services are provided to the citizens, entrepreneurs and public servants on the 24/7 bases. These services are available through domain-specific portals to a variety of user groups (citizens, entrepreneurs, public servants). All Estonian residents having a national ID-card can access these services through X-Road. The number of requests per month exceeds currently 3 million. For brevity we are going to call the whole information system from now on as X-Road.

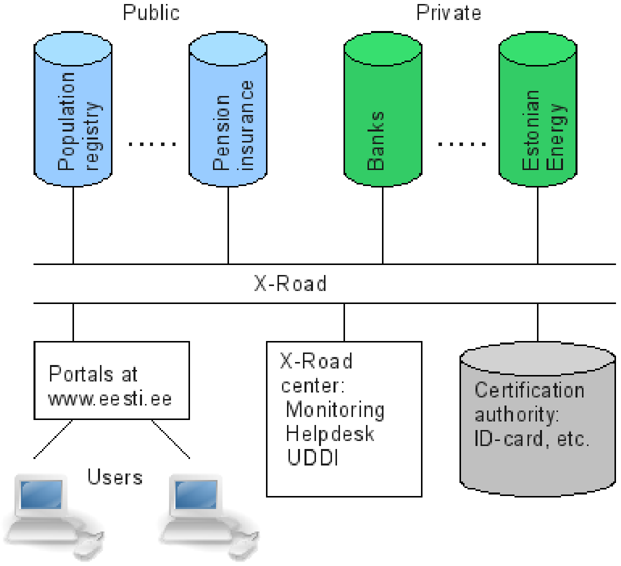


Figure 1. X-Road connects public and private information service providers.

Integration of databases developed by different developers at different times has been a difficult task that started in 2001 and has resulted in a widely used system at the present. During this time a number of standard tools have been developed to enable the creation of e-services capable of simultaneously using the data from different national and international databases. These services enable to read and write data, develop business logic based on data, etc. In X-Road data exchange is handled by SOAP messages, Web services are described in WSDL and service descriptions are published at a UDDI repository.

Figure 1 shows the simplified structure of Estonian information system based on X-Road. As demonstrated in the figure, X-Road connects besides public databases also some private ones. These are, for instance, main banks and some privately owned infrastructure enterprises. Some services are provided by the X-Road infrastructure itself that includes PKI infrastructure, help-desk, monitoring, etc. Users connect to the system through portals where they can execute predefined services. All queries have to be done one by one, even if semantic connections exist between services.

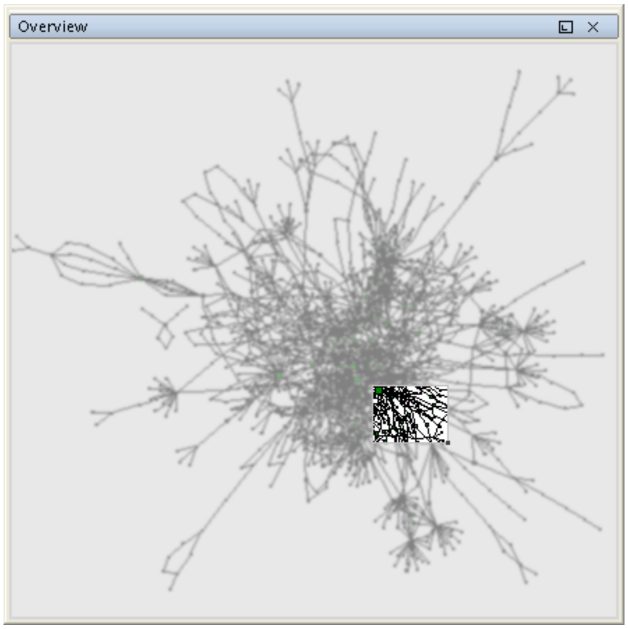


Figure 2. X-Road service model.

This large system is continuously changing and also its maintenance requires often new software updates. When new organization is joining the X-Road it means that they will make their services available through X-Road and/or need to access services offered through the X-Road. In the former case number of new services will be published and in the latter case specialized queries may be needed. This accentuates the problem of automation of composition of services on the service model of X-Road.

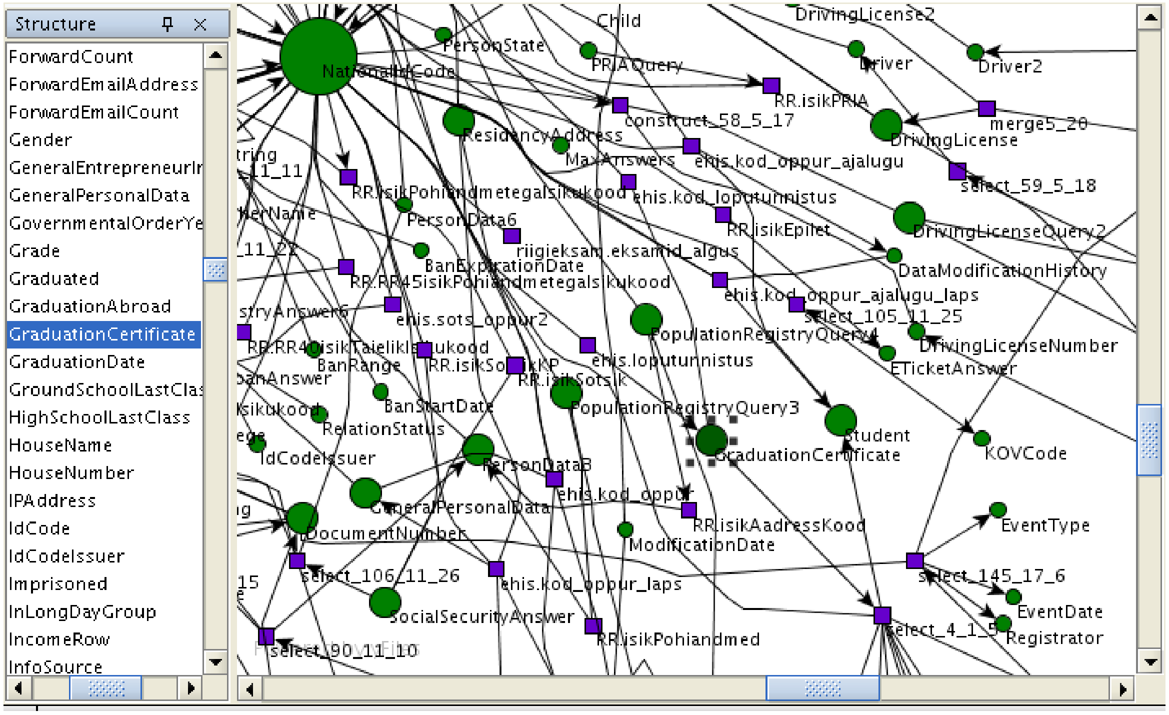


Figure 3. Zoomed-in part of X-Road service model.

3. X-Road service model

Analysis [3] of already operational X-Road resulted in a service model that included about 300 atomic services and about 600 unique references to semantic resources. These services have been annotated and descriptions of their interfaces constitute the syntactic service model shown in Figure 2. This figure shows a visual representation of the whole service model visualized by Java graph editor yEd. It is a large graph where nodes are atomic services and resources.

A small part of the model shown by a rectangle in Figure 2 is enlarged in the in Figure 3. List in the left pane of the user interface fragment shows a scrollable list of all resources. One can see atomic services as rectangles and inputs and outputs as circles (e.g., *GraduationCertificate*, *Student*, etc.) connected to services. The highlighted resource *GraduationCertificate* is input for the atomic services *select_4_1_5* and *ehis.kod_loputunnistus*, and output for the service *ehis.loputunnistus*. This is shown by particular arrows. Size of a circle of a resource shows its relative importance (connectivity to services). A resource with the largest value in the current model is *NationalIdCode* that is not surprising, because it is used in most of the services.

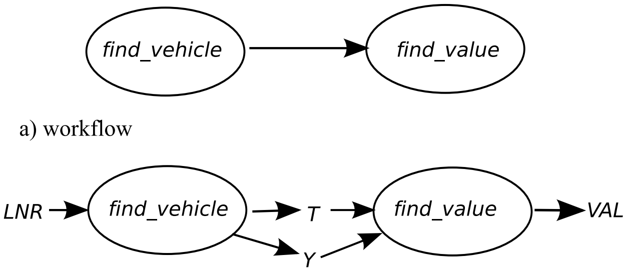
The model presented here is the basis for automatic synthesis of services. However, it must be transformed into another format in order to be applicable as a specification for the synthesis. This format is prescribed by the tool we use. It will be discussed in Section 5 on an example. Before going to explain the tool, we present the logical basis of synthesis in the next section.

4. Automatic handling of a Web service model

In the present section, we describe a logic-based method of automatic composition of services that is the basis of the tool we are using. The method is based on structural synthesis of programs (SSP) [5] and has been in use in several programming tools [4]. In this setting, a service is considered as a computational problem – computing a desired output from a given input. The problem is described for SSP by a set of formulas automatically extracted from a specification, and a goal that the expected result is computable is formulated as a theorem to be proven. A proof of solvability of the problem is built, and a program for solving the problem is extracted from the proof. SSP uses intuitionistic logic that is a constructive logic, i.e., a logic where any proof of existence of an object also is supported by an algorithm that enables one to construct the object [6].

Let us explain the synthesis of services on an example. The problem is to find an estimate of total value of a company's vehicles that we will denote by *RESULT* from the company's official registry number *RNR*. The first and simple task can be to start with a license number *LNR* of a vehicle, find its type *T* and production year *Y*, and to calculate an estimated value *VAL* of a car for *T* and *Y*.

To be able to represent complex services with control structures (loops, choices, etc.) we are using higher order workflow (HOWF) with data dependencies [7]. We translate workflows in a logic that enables us to reason about the reachability of goals on workflow models. Workflow with data dependencies includes explicitly represented inputs and outputs of every atomic service – data items. An ordinary workflow graph and the respective workflow graph with data dependencies for our simple task are shown in Figure 4.



a) workflow

b) workflow with data dependencies

Figure 4. Two representations of a workflow.

In a workflow with data dependencies one does not need extra arrows for representing the order of execution of atomic services, because the data dependencies determine the required order. (If no data are passed between two services and their execution order is still important, then one can add a dummy data item to determine the execution order.)

To solve the whole example problem, one has first to find a list of all vehicles *VEH* from a given registry number *RNR* of a company. Let us call the respective node *all_vehicles*. Thereafter, in order to compute *RESULT*, one has to repeat the task shown in Figure 4 for each vehicle found for the company. This requires introduction of some control in the composed service. This control will be represented by a node called *loop* that will pass *LNR* to the workflow shown in Figure 4 and collect *VAL* as the result of the task performed by this workflow. It is important to note that the data items *LNR* and *VAL* are not input and output of the node *loop*. They are input and output for the part of the workflow that solves a subtask – computes *VAL* from *LNR*, and they are bound with the *loop* node by dotted arrows, see Figure 5. The complete workflow for solving the example problem is shown in Figure 5. The node *loop* is a higher order node – one of its inputs is a subtask "compute *VAL* from *LNR*" and the workflow is a higher order workflow (HOWF). More on using HOWF for representing Web services can be found in [7].

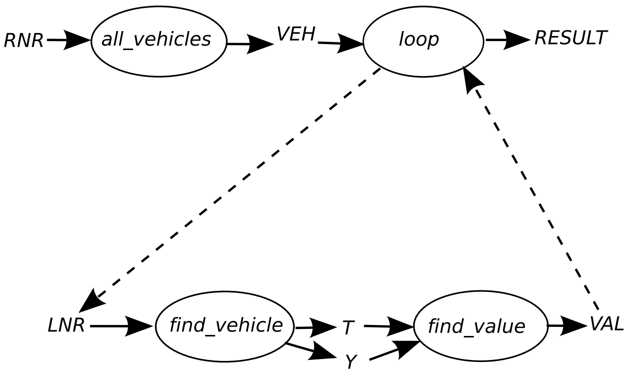


Figure 5. Higher order workflow.

We have not yet discussed the synthesis of services – only their representation by means of HOWF has been described. Now we will show that a HOWF in its turn can be represented in a propositional logic. Let us consider any data name *X* in a workflow as a proposition "there is a way to find the value of *X*". Then a service that computes, for instance, *VEH* from *RNR*, can be encoded in the intuitionistic logic as an implication

$$RNR \supset VEH\{all\_vehicles\},$$

because this implication says "from the fact that there is a way to find the value of *RNR* follows that there is a way to find the value of *VEH*". The name in curly brackets denotes in constructive logic a function that realizes the implication. In our case it is the atomic service *all_vehicles*. The general rule is that a service can be represented by an implication where its inputs are conjuncts on the left side and outputs are conjuncts on the right side. This is also how all services from a service model can be represented in the logic. But we have higher order nodes as well. In this case one has to consider a subtask as an extra input. A subtask itself is represented by an implication, in our example it is

$$LNR \supset VAL\{\varphi\},$$

where it has an unknown realization denoted by a functional variable φ. Adding this implication to inputs of *loop* node, we get the following formula for the *loop*:

$$(LNR \supset VAL)\{\varphi\} \wedge VEH \supset RESULT\{loop(\varphi)\}$$

A logical description of the complete workflow from Figure 5 is the following:

$$\left.\begin{array}{c} RNR \supset VEH\{all\_vehicles\} \\ (LNR \supset VAL)\{\varphi\} \wedge VEH \supset RESULT\{loop(\varphi)\} \\ LNR \supset T \wedge Y\{find\_vehicle\} \\ T \wedge Y \supset VAL\{find\_value\}. \end{array}\right\} (*)$$

In our implementation, the whole service model (hundreds of atomic services) is represented in the logic as above.

$$\dfrac{RNR \supset VEH\{\mathbf{all\_vehicles}\} \qquad \dfrac{(LNR \supset VAL)\{\varphi\} \wedge VEH \supset RESULT\{\mathbf{loop}(\varphi)\} \qquad \dfrac{LNR \supset T \wedge Y\{find\_vehicle\} \quad T \wedge Y \supset VAL\{\mathbf{find\_value}\}}{LNR \supset VAL\{find\_vehicle; find\_value\}}\,(SSP2)}{VEH \supset RESULT\{loop(find\_vehicle; find\_value)\}}\,(SSP1)}{RNR \supset RESULT\{all\_vehicles; loop(find\_vehicle; find\_value)\}}\,(SSP2)$$

Figure 6. Proof of $RNR \supset RESULT$.

When a new composite service with inputs $X_1, \ldots, X_m$ and outputs $Y_1, \ldots, Y_n$ has to be built, a goal is given in the form of an implication

$$X_1 \wedge \ldots X_m \supset Y_1 \wedge \ldots Y_n\{\psi\},$$

where ψ is a functional variable denoting the composite service that has to be found. Let us have the goal

$$RNR \supset RESULT\{\psi\},$$

and assume that the formulas (*) are included in the service model as a part of the model. Then our tool searches for a proof of the goal, and after finding it, translates the proof into required form of the service description. The proof in our case includes only three steps as we see in Figure 6. The first step is deriving $LNR \supset VAL$ from $LNR \supset T \wedge Y$ and $T \wedge Y \supset VAL$ using a rule of structural synthesis denoted by *SSP2*, and described below, etc.

One can see that at deriving a new formula also its realization is built, and the whole program appears gradually step by step: first *find_vehicle; find_value*, then *loop(find_vehicle; find_value)* and finally *ψ=all_vehicles;loop(find_vehicle; find_value)*.

A special feature of this proof is that its every step corresponds to an application of at least one atomic service (shown in bold font in Figure 6). This is achieved due to a special form of inference rules – the SSP rules. These rules are admissible rules of intuitionistic logic, i.e., they enable one to construct only logically correct proofs. From the other side, these rules are very good for proof search, because there is no need to make a separate step for every logical connective (conjunction or implication). A notation of a respective rule is shown at each derivation step. The SSP rules are shown here with metasymbols *A,B,C,D,G,Z,W* denoting conjunctions of propositional variables and *X* denoting a conjunction of propositional variables and implications that represent subtasks:

$$\frac{(A \supset B) \wedge X \supset Z : f \quad A \wedge W \supset B : g}{X \wedge W \supset Z : f(g)} \qquad \text{(SSP1)}$$

$$\frac{A \supset B \wedge C : f \quad B \wedge D \supset G : g}{A \wedge D \supset C \wedge G : f; g} \qquad \text{(SSP2)}$$

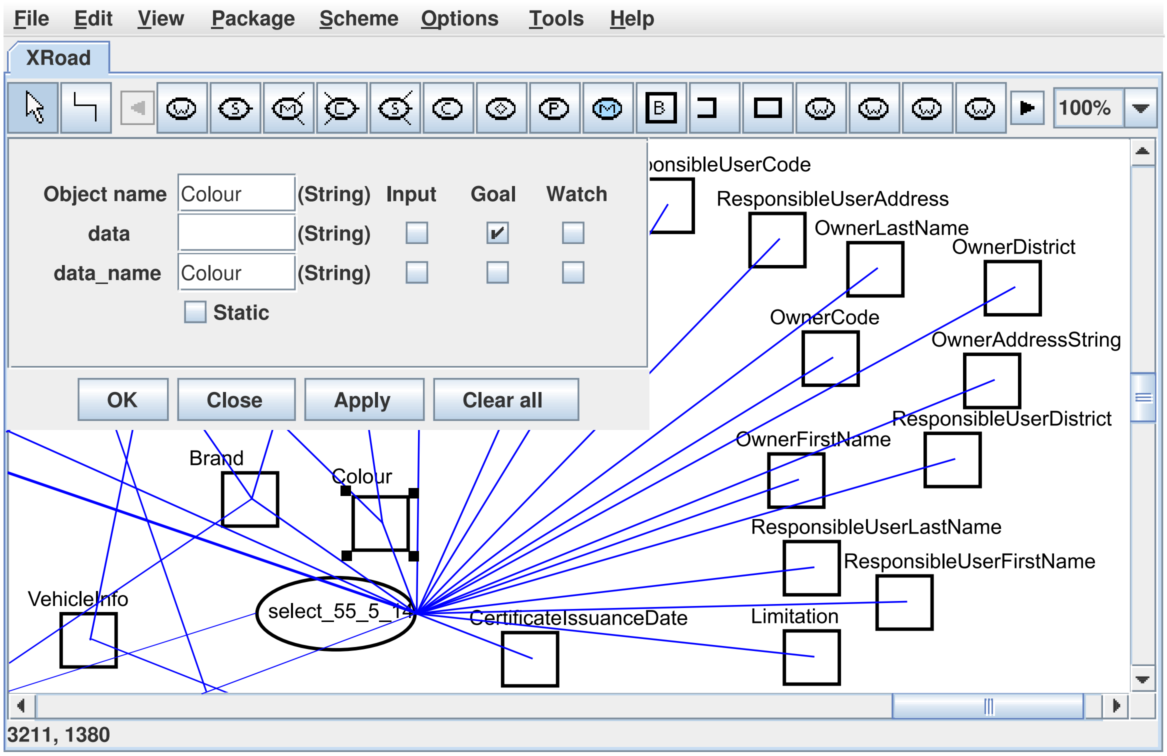
This is a brief explanation of the logic used for synthesis of compositions of atomic services. The tool for automatic composition of services has been implemented in the software environment CoCoViLa that includes an SSP-based algorithm synthesis part and is able to handle specifications given in the visual or textual form [8]. Internal representation of logical formulas in CoCoViLa is not a text, i.e., not formulas as we see them here, but a complex data structure with cross-references, designed with the aim of providing the required performance even in the case of a large number of atomic services. However, the implementation respects precisely the logic explained here. An example in the next section demonstrates the usage of this tool in more detail.

5. Implementation

We have implemented a prototype tool for automatic composition of services in the software environment CoCoViLa that includes an SSP-based algorithm synthesis part and is able to handle specifications given in the visual or textual form.

A visual language for representing Web services, control nodes, data resources and their connections has been developed. By using the visual language, visual service models can be constructed. Known inputs and desired outputs from which the goal is formulated can be defined on the model. A visual model and the goal are automatically translated into a textual specification that is used to generate a Java program if the proof of solvability of the problem can be built. By running the Java program, we can generate BPEL or OWL-S description of the complex service or execute complex service from the Java code.

Figure 7 shows the order of steps that are done by our composition tool in order to compose a new complex service. To use the composition system, developer of new complex service has to know how to define the goal, i.e., desired outputs and needed inputs of the complex service. After inputs and outputs have been defined, a service composition algorithm will be synthesized automatically. The structure of this algorithm already represents the structure of the complex service to be generated. However, CoCoViLa



			Input	Goal	Watch
Object name	Colour	(String)			
data		(String)	□	☑	□
data_name	Colour	(String)	□	□	□

□ Static

3211, 1380

Figure 8. Specifying a compound X-Road service on CoCoViLa service model.

produces only a Java code, but we need a representation in BPEL or OWL-S. Therefore the steps of the code use preprogrammed generators of BPEL or OWL-S. When this code is run, a service description corresponding to the structure of the synthesized algorithm is generated using additional information from the initial specification.

All intermediate steps, that is steps between defining a goal on the model and getting complex service description as an output, are done automatically by the composition tool. However, if necessary, intermediate steps (e.g., a structure of the complex service or a generated Java code) can be visualized for the developer, for instance, for debugging purposes.

We have visualized a syntactic service model for X-Road services with our service composition tool. Model used in our composition tool is generated from the one described in Section 1. Figure 8 shows a small part of the X-Road model and data properties window for a resource. Services are represented by ovals and data resources by squares in our visual language.

To illustrate the composition process described in previous section, let us consider a task where an official has to identify graduate's home address, occupation area and some attributes of its car, e.g., license number and color of the car. Input is the graduation certificate of the person. Getting a required service includes the following manual steps.

1. Using an ontology/dictionary, find the name of the considered data items in the service model. These names in the present case happen to be the following: *GraduationCertificate*, *EstonianAddressString*, *OccupationArea*, *RegistrationMark*, *Colour*. Note that these data items belong to different databases managed by different organizations.

2. Mark the input (*GraduationCertificate*) and requested outputs (*EstonianAddressString*, *RegistrationMark*, *Colour*, *OccupationArea*) on a visual representation of the service model.

3. Define the settings for formatting output (e.g., BPEL) and some information related to the generated complex service, for instance, its name, filename the service description is written to, etc. The rest will be done by our composition tool.

A developer can define output by marking it to be a goal as shown in the Figure 8. The model can be zoomed in and adjusted for more detailed analysis. Composition tool includes a search window, to ease the finding of data resources

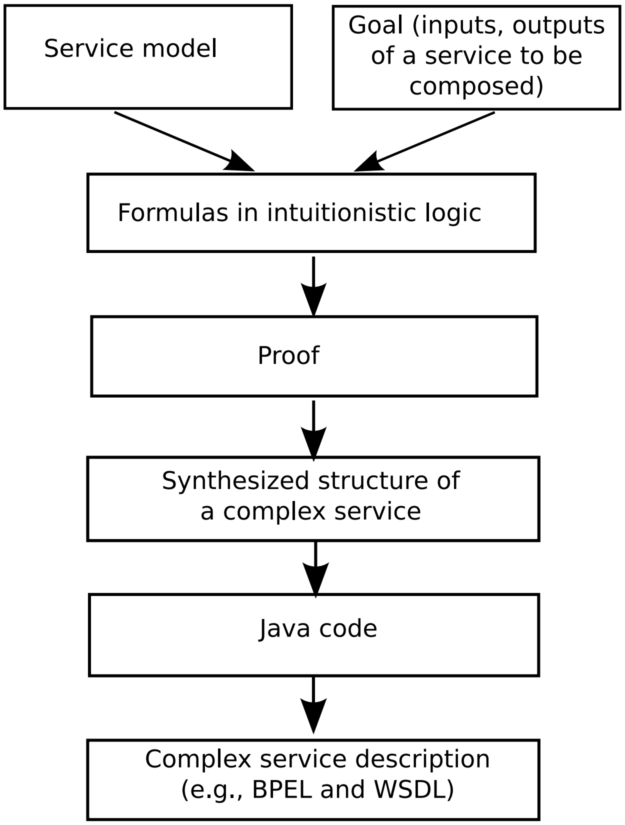

Service model
Goal (inputs, outputs of a service to be composed)
Formulas in intuitionistic logic
Proof
Synthesized structure of a complex service
Java code
Complex service description (e.g., BPEL and WSDL)

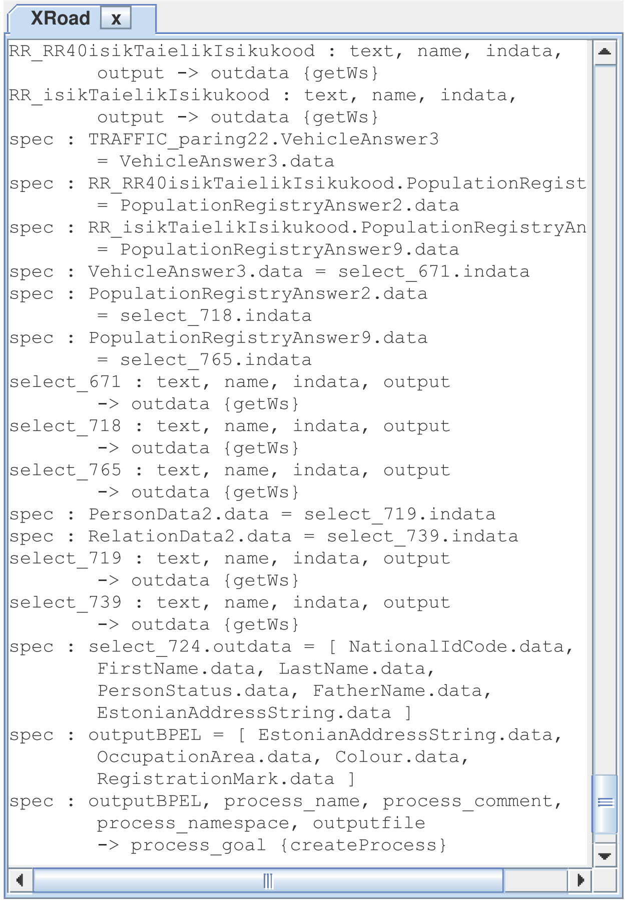
Figure 7. Automatic steps in composition process.

or services. If a goal of getting output data from the input data is provable, a complex service and its specification in BPEL will be generated.

The proof of reachability of the goal is constructed automatically by CoCoViLa as described in Section 4. The proof obtained gives the structure of the service to be constructed, this is also an algorithm to construct the service. In the present example, the algorithm includes 1434 lines, and part of it is visualized by CoCoViLa in a separate window shown in Figure 9. Algorithm gives an order in which services should be executed. Lines starting with service name (e.g., *RR_RR40isikTaielikIsikukood*, *select_739*), are representing services that need to be invoked in order to compute the goal. After the service name there is an implication and a function implementing it:

$$text, name, indata, output- > outdata\{\texttt{getWs}\}.$$

This line specifies that having an input (*text*, *name*, *indata*, *output*), we know that output – *outdata*, is computable using a function `getWS`. Functions referred to from the algorithm are Java functions that will be used in Java code and are executed when the code is run.



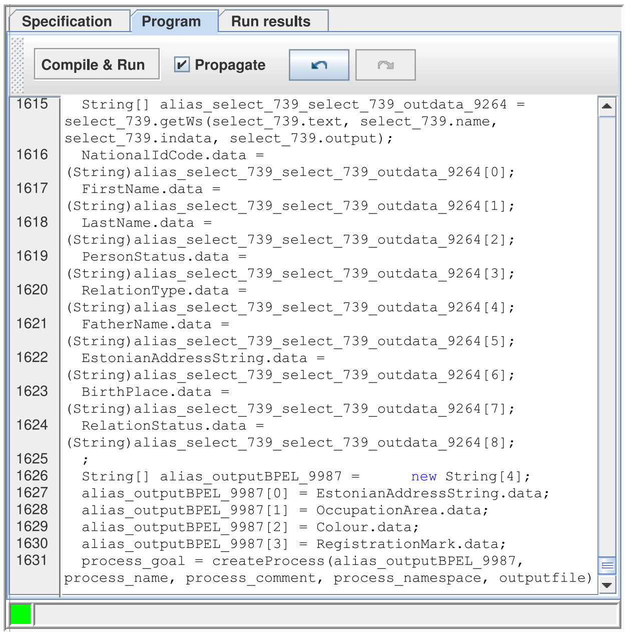
```
RR_RR40isikTaielikIsikukood : text, name, indata,
        output -> outdata {getWs}
RR_isikTaielikIsikukood : text, name, indata,
        output -> outdata {getWs}
spec : TRAFFIC_paring22.VehicleAnswer3
        = VehicleAnswer3.data
spec : RR_RR40isikTaielikIsikukood.PopulationRegist
        = PopulationRegistryAnswer2.data
spec : RR_isikTaielikIsikukood.PopulationRegistryAn
        = PopulationRegistryAnswer9.data
spec : VehicleAnswer3.data = select_671.indata
spec : PopulationRegistryAnswer2.data
        = select_718.indata
spec : PopulationRegistryAnswer9.data
        = select_765.indata
select_671 : text, name, indata, output
        -> outdata {getWs}
select_718 : text, name, indata, output
        -> outdata {getWs}
select_765 : text, name, indata, output
        -> outdata {getWs}
spec : PersonData2.data = select_719.indata
spec : RelationData2.data = select_739.indata
select_719 : text, name, indata, output
        -> outdata {getWs}
select_739 : text, name, indata, output
        -> outdata {getWs}
spec : select_724.outdata = [ NationalIdCode.data,
        FirstName.data, LastName.data,
        PersonStatus.data, FatherName.data,
        EstonianAddressString.data ]
spec : outputBPEL = [ EstonianAddressString.data,
        OccupationArea.data, Colour.data,
        RegistrationMark.data ]
spec : outputBPEL, process_name, process_comment,
        process_namespace, outputfile
        -> process_goal {createProcess}
```

Figure 9. Synthesized structure of a complex service.

Some checks about availability of relevant information (service name, output filename, etc.) necessary for composed service description are also done. This is shown in the last line (starting with *spec*), which shows that given goal is provable (this is indicated with *process_goal* as output), if the following inputs are given: text generated so far (*outputBPEL*) and information about complex service (*process_name*, *process_comment*, *process_namespace*). To compute the goal, method *createProcess* will be used. Lines starting with *spec* contain assignments that need to be done in order to prove the goal.

Java code is extracted from the synthesized structure of the complex service and data from the initial model (e.g., grounding of services and requested composition description language). Screenshot of a piece of the Java code made visible to the service developer can be seen in Figure 10. Program includes more than 1600 lines of code.

Extracted Java code is not the code of complex service, but a complex service description generator. Functions shown in curly brackets, e.g., `getWs`, as realizations of implications in Figure 9 are part of the generated Java code. See for example line number 1615, where object *select_739* has function `getWs`, with arguments that were given as input for the implication shown in Figure 9. Function `getWs` is taking BPEL construct given in *text*, includes information specific for the given service (given in *name*, *indata* and *output*) and adds generated construct to the BPEL output generated before. Similarly *process_goal* will be computed with `createProcess` function. This is shown in line 1631.



```
1615   String[] alias_select_739_select_739_outdata_9264 =
select_739.getWs(select_739.text, select_739.name,
select_739.indata, select_739.output);
1616   NationalIdCode.data =
(String)alias_select_739_select_739_outdata_9264[0];
1617   FirstName.data =
(String)alias_select_739_select_739_outdata_9264[1];
1618   LastName.data =
(String)alias_select_739_select_739_outdata_9264[2];
1619   PersonStatus.data =
(String)alias_select_739_select_739_outdata_9264[3];
1620   RelationType.data =
(String)alias_select_739_select_739_outdata_9264[4];
1621   FatherName.data =
(String)alias_select_739_select_739_outdata_9264[5];
1622   EstonianAddressString.data =
(String)alias_select_739_select_739_outdata_9264[6];
1623   BirthPlace.data =
(String)alias_select_739_select_739_outdata_9264[7];
1624   RelationStatus.data =
(String)alias_select_739_select_739_outdata_9264[8];
1625   ;
1626   String[] alias_outputBPEL_9987 =      new String[4];
1627   alias_outputBPEL_9987[0] = EstonianAddressString.data;
1628   alias_outputBPEL_9987[1] = OccupationArea.data;
1629   alias_outputBPEL_9987[2] = Colour.data;
1630   alias_outputBPEL_9987[3] = RegistrationMark.data;
1631   process_goal = createProcess(alias_outputBPEL_9987,
process_name, process_comment, process_namespace, outputfile)
```

Figure 10. Extracted Java program.

The final step of the composition is generation of the Web service description in the final form, e.g., in BPEL as requested in the example. This is done by compiling the synthesized Java code and running it. Figure 11 shows a fragment of the BPEL output (about 50 lines totally) of the generated service. Figure includes first half of the BPEL sequence with nine service invocations to different databases (four of which are shown), that need to be done in order to satisfy the simple goal that we were using as an example. In addition to BPEL sequence, represented in the figure, it is possible to generate other BPEL constructs (e.g., *while*, *condition*). It is also possible to create generators for other languages. So far we have only experimented with BPEL and OWL-S. It is important to notice that the steps of proving, compiling and service text generation are performed automatically, without interference of the user.

The synthesis algorithm has linear time complexity and can be applied to very large syntactic models. The time spent for solving the example here was about one second on a laptop with 1.2 GHz Intel processor.

```
...
<sequence>
<receive createInstance="yes" name="start"
operation="getComplexService"
partnerLink="XRoadClientPL"
portType="wsdl:XRoadClientP"
variable="Request" />
<invoke name="select_4_1_5"
partnerLink="XRoadClientPL"
operation="select_4_1_5"
inputVariable="getselect_4_1_5"
outputVariable="select_4_1_5Response"/>
<invoke name="TRAFFIC_paring22"
partnerLink="XRoadClientPL"
operation="TRAFFIC_paring22"
inputVariable="getTRAFFIC_paring22"
outputVariable="TRAFFIC_paring22Response"/>
<invoke name="RR_RR40isikTaielikIsikukood"
partnerLink="XRoadClientPL"
operation="RR_RR40isikTaielikIsikukood"
inputVariable=
"getRR_RR40isikTaielikIsikukood"
outputVariable=
"RR_RR40isikTaielikIsikukoodResponse"/>
<invoke name="RR_isikTaielikIsikukood"
partnerLink="XRoadClientPL"
operation="RR_isikTaielikIsikukood"
inputVariable="getRR_isikTaielikIsikukood"
outputVariable=
"RR_isikTaielikIsikukoodResponse"/>
...
```

Figure 11. BPEL fragment of a composed service.

6. Related work

A number of methods for dynamic composition of Web services have been proposed since the introduction of Web services standards. Majority of them fall into one of the following two categories: methods based on pre-defined workflow models and methods, which build the workflows from scratch. For the methods in the first category, the user should specify the workflow of the required composite service, including both nodes and the control flow and the data flow between the nodes. The nodes are regarded as abstract services that contain search templates. The concrete services are selected and bound at runtime according to the search recipes (see [9] and [10], for instance).

The second category includes methods related to AI planning, automated theorem proving, graph search, etc. They are based on the assumption that each Web service is an action which alters the state of the world as a result of its execution. Since Web services (actions) are software components, the input and the output parameters of Web services act as preconditions and effects in the planning context. After a user has specified inputs and outputs required by the composite service, a workflow (plan) is generated automatically by AI planners or other tools from the scratch.

Theoretically any domain-independent AI planner can be applied for Web service composition. In [11] SHOP2 planner is applied for automatic composition of DAML-S services. Other planners, which have been applied for automated Web service composition, include [12],[13],[14],[15], just to mention a few of them.

Waldinger [16] proposes initial ideas for a deductive approach for Web services composition. The approach is based on automated deduction and program synthesis and has its roots in the work presented in [17]. Initially available services and user requirements are described with a first-order language, related to classical logic, and then constructive proofs are generated with Snark [18] theorem prover. From these proofs workflows can be extracted.

Although only conjunctions are allowed for describing services and user requirements in most cases of deductive composition methods, Lämmermann [19] takes advantage of disjunctions in intuitionistic logic as well. Disjunctions are used to describe exceptions, which may be thrown during service invocations.

McDermott [20] tackles the closed world assumption in AI planning while composing Web services. He introduces a new type of knowledge, called value of an action, which allows modeling resources or newly acquired information – entities, which until this solution were modeled extra-logically. Anyway, while using resource-conscious logics, like linear logic, applied by Rao et al [21], or transition logic, this problem is treated implicitly and there is no need to distinguish informative and truth values. Since linear logic is not based on truth values, we can view generated literals as references to informative objects.

Hull and Su [22] present a short overview of tools and models for Web service composition. The models include OWL-S, the Roman model [23] and the Mealy machine [24]. While OWL-S includes a rich model of atomic services and how they interact with an abstraction of the "real world", the Roman model and the Mealy machine use a finite state automata framework for representing workflows.

Hashemian and Mavaddat [25] combine breadth-first graph search and interface automata [26] for automating Web service composition. While graph search is used for finding a path with minimum length from identified input nodes to identified output nodes, interface automata is applied for composing paths into a composite Web services. Graph search operates over a directed graph, where edges represent available Web services and nodes represent inputs/outputs of particular Web services.

Although there is enormous amount of literature available describing different composition methods and methodologies, not so many graphical composition environments such as CoCoViLa have been described and implemented so far. Sirin et al [27] propose a semi-automatic Web service composition scheme for interactively composing new Semantic Web services. Each time a user selects a new Web service, the Web services, that can be attached to inputs and outputs of the selected service, are presented to the user. Much manual search is avoided in this way. The process could be fully automated by applying our methodology if user requirements to the resulting service are known a priori. CoCoViLa complements this tool by providing a GUI for supporting specification of user requirements and synthesis of solutions based on them.

Gómez-Pérez et al [28] describe another graphical tool for Semantic Web service composition. This tool enables the user to specify graphically the input/output interactions among the sub-services that constitute the required service. Once the design has been checked, wrappers perform the translations from the instances of framework ontologies into the OWL-S specification.

Rao et al [29] describe a tool for mixed initiative framework for semantic Web service discovery and composition that aims at flexibly interleaving human decision making and automated functionality in environments where annotations may be incomplete and even inconsistent. An initial version of this framework has been implemented in SAPs Guided Procedures, a key element of SAPs Enterprise Service Architecture (ESA). This is a graphical tool for aiding composition if no or only partial semantic annotations of Web services are given.

Hakimpour et al [30] present a tool based on the model that supports a user-guided interactive composition approach, by recommending component Web services according to the composition context. This tool is based on a model for composition of Web services, which complements the WSMO orchestration in IRS-III – a framework for semantic Web services based on WSMO specification.

Since service composition is an application and an integral part of semantic interoperability, we shall enlist here some of these initiatives as well. In addition to German initiative Deutschland Online [31], Italian initiative in public administration [32] there are Finnish semantic initiative FinnONTO [33], and Semantic Latvia project [34]. The scope and accent of these initiatives are quite different – some focus on consolidating semantic assets in several governmental institutions already in place into semantic portals, some on building full-scale national semantic web infrastructures, others target syntactic or semantic descriptions of data schemas, some are on the level of human-oriented descriptions of assets, others try to reach automatic use. Estonian semantic interoperability initiative [35] is focused on providing semantic descriptions of public Web services and thus provides a valuable source for evaluating automated composition tools in large scale.

There are also pan-European initiatives, which include SEMIC (SEMantic Interoperability Centre Europe) [36], led by the European Commissions IDABC program [37], and semanticGov [38]. This shows a real need for automation of composition of governmental Web services.

7. Concluding remarks

We have shown in the present work the feasibility of automatic composition of Web services on a very large syntactic service model of governmental services. This approach can be used without any changes for composition of services for large companies as soon as a federated syntactic service model can be built. It could be useful, first of all, for software developers who are extending and modifying a large existing information system that in the present case provides Web-based services for citizens. The advantages of this approach are, first, provable ease of introduction of new services and, second, guaranteed correctness of the services. This approach is scalable to very large service models as can be seen from the synthesis time of services and space requirements of the service model. The available tool supports easy maintenance of the service model – it can be modified on the fly.

In principle, the described tool, after adjusting its user interface, could be given to end users, so that they could develop wanted services by themselves. However, in the present form it is impossible, because it would require more skills that an average citizen has. Making composition of services publicly available may include also security risks. Our experience shows that the main difficulty that a user would have is the unsolvability of the synthesis task – more inputs will be needed than given in the initial service description. This is a semantic debugging problem where partial evaluation could give some help. The partial evaluation on models similar to syntactic service models has been investigated in [39]. The presented work concerns only stateless services. In the case of stateful services, several services have to work simultaneously and have to be orchestrated respectively. Experiments based on CoCoViLa support this by means of higher-order service schemas [7]. However, in the present work the services are stateless, and we do not use this feature.

Acknowledgments

This work is partially supported by the Estonian Science Foundation Grant No. 6886 and Estonian Information Technology Foundation's Tiger University program.

References

[1] R. Maigre, P. Küngas, M. Matskin, and E. Tyugu. Handling large Web services models in a federated governmental information system. In *ICIW '08: Proceedings of the 2008 Third International Conference on Internet and Web Applications and Services*, pages 626–631, Washington, DC, USA, 2008. IEEE Computer Society.

[2] Information technology in public administration of Estonia. Yearbook 2006, Estonian Ministry of Economic Affairs and Communication, 2007.

[3] P. Küngas and M. Matskin. From Web services annotation and composition to web services domain analysis. *International Journal of Metadata, Semantics and Ontologies*, 2(3):157–178, 2007.

[4] P. Grigorenko, A. Saabas, and E. Tyugu. Visual tool for generative programming. *ACM SIGSOFT Software Engineering Notes*, 30(5):249–252, 2005.

[5] G. Mints and E. Tyugu. Justifications of the structural synthesis of programs. *Sci. Comput. Program.*, 2(3):215–240, 1982.

[6] S. C. Kleene. *Introduction to metamathematics*. Elsevier, 1980.

[7] M. Matskin, R. Maigre, and E. Tyugu. Compositional logical semantics for business process languages. In *Proceedings of Second International Conference on Internet and Web Applications and Services (ICIW 2007)*. IEEE Computer Society, 2007.

[8] M. Matskin and E. Tyugu. Strategies of structural synthesis of programs and its extensions. *Computing and Informatics*, 20:1–25, 2001.

[9] F. Casati, S. Ilnicki, L.-J. Jin, V. Krishnamoorthy, and M.-C. Shan. Adaptive and dynamic service composition in eFlow. In *Proceedings of 12th International Conference on Advanced Information Systems Engineering (CAiSE 2000)*, volume 1789 of Lecture Notes in Computer Science, pages 13–31. Springer-Verlag, 2000.

[10] H. Schuster, D. Georgakopoulos, A. Cichocki, and D. Baker. Modeling and composing service-based and reference process-based multi-enterprise processes. In *Proceeding of 12th International Conference on Advanced Information Systems Engineering (CAiSE 2000)*, volume 1789 of Lecture Notes in Computer Science, pages 247–263. Springer-Verlag, 2000.

[11] D. Wu, B. Parsia, E. Sirin, J. Hendler, and D. Nau. Automating DAML-S Web services composition using SHOP2. In *Proceedings of the 2nd International Semantic Web Conference (ISWC 2003)*, 2003.

[12] M. Pistore, F. Barbon, P. Bertoli, D. Shaparau, and P. Traverso. Planning and monitoring web service composition. In *Proceedings of the 11th International Conference on Artificial Intelligence, Methodologies, Systems, and Applications (AIMSA 2004)*, volume 3192 of Lecture Notes in Computer Science, pages 106–115. Springer-Verlag, 2004.

[13] M. Pistore, P. Traverso, P. Bertoli, and A. Marconi. Automated synthesis of composite BPEL4WS Web services. In *Proceedings of 2005 IEEE International Conference on Web Services (ICWS 2005)*, pages 293–301, 2005.

[14] M. Sheshagiri, M. desJardins, and T. Finin. A planner for composing services described in DAML-S. In *Proceedings of the AAMAS Workshop on Web Services and Agent-based Engineering*, 2003.

[15] P. Traverso and M. Pistore. Automated composition of semantic web services into executable processes. In *Proceedings of 3rd International Semantic Web Conference (ISWC 2004)*, volume 3298 of Lecture Notes in Computer Science, pages 380–394. Springer-Verlag, 2004.

[16] R. Waldinger. Web agents cooperating deductively. In *Proceedings of FAABS 2000*, volume 1871 of Lecture Notes in Computer Science, pages 250–262. Springer-Verlag, 2000.

[17] Z. Manna and R. J. Waldinger. A deductive approach to program synthesis. *ACM Transactions on Programming Languages and Systems*, 2(1):90–121, 1980.

[18] M. Stickel, R. Waldinger, M. Lowry, T. Pressburger, and I. Underwood. Deductive composition of astronomical software from subroutine libraries. In *Proceedings of 12th International Conference on Automated Deduction (CADE 1994)*, volume 814 of Lecture Notes in Artificial Intelligence, pages 341–355. Springer-Verlag, 1994.

[19] S. Lämmermann. *Runtime service composition via logic-based program synthesis*. PhD thesis, Department of Microelectronics and Information Technology, Royal Institute of Technology, Stockholm, 2002.

[20] D. McDermott. Estimated-regression planning for interaction with Web services. In *Proceedings of the 6th International Conference on AI Planning and Scheduling*. AAAI Press, 2002.

[21] J. Rao, P. Küngas, and M. Matskin. Composition of semantic Web services using linear logic theorem proving. *Information Systems, Special Issue on the Semantic Web and Web Services*, 31(4-5):340–360, 2006.

[22] R. Hull and J. Su. Tools for composite Web services: A short overview. *SIGMOD Record*, 34(2):86–95, 2005.

[23] D. Berardi, D. Calvanese, G. de Giacomo, M. Lenzerini, and M. Mecella. Automatic composition of e-services that export their behavior. In *Proceedings of the First International Conference on Service-Oriented Computing (ICSOC 2003)*, volume 2910 of Lecture Notes in Computer Science, pages 43–58. Springer-Verlag, 2003.

[24] T. Bultan, X. Fu, R. Hull, and J. Su. Conversation specification: A new approach to design and analysis of e-service composition. In *Proceedings of 12th International World Wide Web Conference (WWW 2003)*, pages 403–410, 2003.

[25] S. V. Hashemian and F. Mavaddat. A graph-based approach to Web services composition. In *Proceedings of 2005 IEEE/IPSJ International Symposium on Applications and the Internet (SAINT 2005)*, pages 183–189. IEEE Computer Society, 2005.

[26] L. de Alfaro and T. A. Henzinger. Interface automata. In *Proceedings of the 8th European Software Engineering Conference held jointly with 9th ACM SIGSOFT International Symposium on Foundations of Software Engineering(ESEC 2001)*, pages 109–120. ACM Press, 2001.

[27] E. Sirin, B. Parsia, and J. Hendler. Composition-driven filtering and selection of Semantic Web services. In *Proceedings of the First International Semantic Web Services Symposium, AAAI 2004 Spring Symposium Series*, pages 129–136. AAAI Press, 2004.

[28] A. Gómez-Pérez, R. Gonzalez-Cabero, and M. Lama. A framework for design and composition of Semantic Web services. In *In Proceedings of the First International Semantic Web Services Symposium, AAAI 2004 Spring Symposium Series*, pages 113–120. AAAI Press, 2004.

[29] J. Rao, D. Dimitrov, P. Hofmann, and N. Sadeh. A mixed initiative approach to semantic web service discovery and composition: Sap's guided procedures framework. In *ICWS '06: Proceedings of the IEEE International Conference on Web Services*, pages 401–410, Washington, DC, USA, 2006. IEEE Computer Society.

[30] F. Hakimpour, D. Sell, L. Cabral, J. Domingue, and E. Motta. Semantic web service composition in irs-iii: The structured approach. In *CEC '05: Proceedings of the Seventh IEEE International Conference on E-Commerce Technology*, pages 484–487, Washington, DC, USA, 2005. IEEE Computer Society.

[31] Deutschland Online. http://www.deutschland-online.de/DOL_en_Internet/broker.jsp. [May 15, 2009].

[32] Italian initiative in public administration. http://www.cnipa.gov.it/site/it-IT/. [May 15, 2009].

[33] E. Hyvönen, K. Viljanen, J. Tuominen, and K. Seppälä. Building a national semantic web ontology and ontology service infrastructure -The FinnONTO approach. In *The Semantic Web: Research and Applications, 5th European Semantic Web Conference, ESWC 2008, Tenerife, Canary Islands, Spain, June 1-5, 2008*, pages 95–109, 2008.

[34] G. Barzdins, R. Balodis, K. Cerans, A. Kalnins, M. Opmanis, and K. Podnieks. Towards semantic Latvia. In *Communications of 7th International Baltic Conference on Databases and Information Systems*, pages 203–218. Technika, 2006.

[35] H.-M. Haav, A. Kalja, P. Küngas, and M. Luts. Ensuring large-scale semantic interoperability: The estonian public sector's case study. In H.-M. Haav and A. Kalja, editors, *Databases and Information Systems V – Selected Papers from the Eighth International Baltic Conference, DB&IS 2008*, pages 117 – 128. IOS Press, 2008.

[36] SEMantic Interoperability Centre Europe. http://www.semic.eu/semic/view/snav/About_SEMIC.xhtml. [May 15, 2009].

[37] European Commission's IDABC program. http://ec.europa.eu/idabc/. [May 15, 2009].

[38] T. Vitvar, M. Kerrigan, A. van Overeem, V. Peristeras, and K. Tarabanis. Infrastructure for the semantic pan-european e-government services. In *Proceedings of the 2006 AAAI Spring Symposium on The Semantic Web meets eGovernment (SWEG)*, 3 2006.

[39] P. Küngas and M. Matskin. Detection of missing Web services: The partial deduction approach. *Special Issue on Recent Innovations in Web Services Practices, International Journal of Web Services Practices*, 1(1-2):133–141, 2005.

Gradual Adaption Model for Information Recommendation Based on User Access Behavior

Jian Chen
Graduate School of Human Sciences
Waseda University
Tokorozawa, Japan
wecan_chen@fuji.waseda.jp

Roman Y. Shtykh
Graduate School of Human Sciences
Waseda University
Tokorozawa, Japan
roman@akane.waseda.jp

Qun Jin
Faculty of Human Sciences
Waseda University
Tokorozawa, Japan
jin@waseda.jp

Abstract—**In this study, we propose a gradual adaption model for information recommendation. This model is based on a set of concept classes that are extracted from Wikipedia categories and pages. Using the extracted information, data representing the users' information access behavior is collected by a unit of one day for each user, and analyzed in terms of short, medium, long periods, and by remarkable and exceptional categories. The proposed model is then established by analyzing the pre-processed data based on Full Bayesian Estimation. We further present experimental simulation results, and show the operability and effectiveness of the proposed model.**

Keywords-information recommendation; data mining; gradual adaption; Wikipedia

I. INTRODUCTION

Today we are surrounded by a *plethora* of information. Except traditional information in books, a variety of web resources are connected with each other by the Internet. We can use search engines to search such information, but the problem is that we cannot retrieve and perceive all search results that are above a number of thousands.

Efficient use of web resources is an important issue we try to resolve. In this study, we propose an information recommendation model called Gradual Adaption Model (GAM) [1]. In this model, we build a set of concept classes that are extracted from Wikipedia categories and pages.

In fact, more and more people are becoming increasingly accustomed to use Wikipedia to find knowledge since 2001. Furthermore, recently Wikipedia articles become more and more often referred by scientific papers. Owing to its good quality and reliability, Wikipedia can also be considered as a resource for information recommendation. Based on this, we investigated Wikipedia, and found that its category structure can be used for extracting a set of concept classes that are used as a classification criterion in our proposed model.

When users access web pages through the system that is built with the proposed GAM, it classifies these Web pages by concept classes. Such user access data are collected by a unit of one day for each user. Based on the collected data, the reuse probability of each concept class is estimated in terms of short, medium, and long periods by Full Bayesian Estimation. If a concept class belongs to more than two periods, it is classified as a concept class of remarkable category. If a concept class is accessed just occasionally, it means its probability is so low that the concept class almost impossibly appears in the front of recommendation results, it is classified as a concept class of exceptional category. When users access web resources next time, GAM will gradually adapt to the transition of users' selection, and recommend web pages for users, according to the concept class probability that is estimated by GAM.

This paper is organized as follows. In Section II, related works are introduced. Section III gives a brief introduction on Full Bayesian Estimation that we apply in this study. A detailed description on GAM is provided in Section IV, and simulation results are discussed in Section V. Finally, Section VI concludes this study and directs future works.

II. RELATED WORKS

As we know, Wikipedia is an open resource that can be modified by anyone. Therefore, we have to face a number of problems such as its reliability and trustworthiness. Further, we will consider these issues by overviewing the works dedicated to Wikipedia, and discuss several modern information recommendation approaches.

A. Wikipedia Information Resource

Nowadays, *an enormous amount* of useful resources can be discovered from Wikipedia. The report by Kashihana et al. [2] shows: there are more than 2.1 million items of the English version recorded in Wikipedia by January 2008. While the number of English articles in Encyclopaedia Britannica (2008 version) is more than 75 thousand. The report also says that the accuracy of articles in Wikipedia and those in the Encyclopaedia is almost equal.

Today, Wikipedia data set, its content and structure are widely used for extracting metadata for research. Wikipedia was found to have an impressive coverage of contemporary documents. As found by Milne et al. [3] after comparing Wikipedia articles and links with a manually-created professional thesaurus, it is a good source of hierarchical and associative relations, with good coverage and accuracy for many areas. Therefore, we can consider Wikipedia categories and theirs pages as an alternative for creation of concept classes and theirs representative indices, which are extracted from Wikipedia categories and theirs pages.

And for the above reasons, mining Wikipedia attracts many researchers. For instance, Mihalcea and Csomai [4] consider the abundance of links embedded in Wikipedia pages and try to extract keywords automatically from them. In addition to embedded links (that can be further classified as incoming links and outcoming links), section headings, template items of Wikipedia pages are considered as semantic features and used to represent a page. The similarity of two Wikipedia pages sharing these features can be used as a page similarity measure [5]. Obviously, the level of representativeness of a term used in a title, headline and text of an article differs. The keywords that occur in the title, headlines and embedded links are better representatives of pages, therefore, they gain higher-weighted values.

An attempt to find good quality articles of Wikipedia in order to recommend them automatically is described by Thomas and Sheth [5]. The idea is to analyze the change of Wikipedia pages by semantic convergence and estimate if these are good articles. This approach can find good articles in Wikipedia, but, in our opinion, to achieve better results and user satisfaction from recommendations, it is important to consider users' needs and behaviors during the information recommendation process.

B. Information Recommendation

Recently, information recommendation is a focus, and attracts much attention by a lot of users and researchers. The web mining [7, 8] approaches have been extensively used for information recommendation. Generally, web mining has been divided into three main areas: usage mining, structure mining, and content mining [7]. As an additional area, semantic web mining [9] was proposed. The following are the data types that are found in the web and mined by these approaches.

- Content data: The text and multimedia data in web pages. It is the real data that is designed and provided to users of a web site.
- Structure data: The data consist of organization inside a web page, internal and external links, and the web site hierarchy.
- Usage data: The web site access logs data.
- User profile: The information data of users. It includes both of data provided by users and data created by the web site.
- Semantic data: The data describe the structure and definition of a semantic web site.

Although web mining is divided into four areas, but they are associated mostly each other, not exclusively.

We recognize the importance of such web mining as content, structure and user profile. In this study, we focus on WUM (Web Usage Mining). In this area, a new document representation model [8] was presented recently. This model is based on implicit users' feedback to achieve better results in organizing web documents, such as clustering and labeling. This model was experimented on a web site with small vocabulary and specific to certain topics. Identifying Relevant Websites from User Activity [10] is another attempt of organizing web pages. It is also based on implicit users' feedback but faces the following problem – to improve retrieval accuracy. The model needs to spend more time to train the system.

However, using implicit users' feedback has such a problem: although there is a relation between the users' implicit feedback, there is also a possibility that a chanciness of implicit users' feedback can impair the relation between web documents clicked by users. Despite this problem, the mining of implicit users' feedback enables us to realize personalized information recommendation. In our work we focus on the implicit feedback coming from the same user, and do not consider interrelation of implicit feedback of different users.

We also noticed that due to the explosive growth of information on the web, web personalization has gained great momentum both in the research and commercial areas [11]. This fact encourages us to use implicit users' feedback to personalize information recommendation.

Dynamic Link Generation [12] is one of early WUM systems. It consists of off-line and on-line modules. In the off-line module, pre-processor extracts information from user access logs and generates records, then clusters the records to categories. The on-line module is used to classify user session records and identify the top matching categories, then return the links that belong to the identified categories to the user.

SUGGEST 3.0 [13] is another kind of WUM systems, but it has only the on-line component. In SUGGEST3.0, the off-line job, like that in Dynamic Link Generation, was realized in the on-line component dynamically. The aim of SUGGEST 3.0 is to manage large web sites. But the size of access logs used to evaluate the system is small and limited.

LinkSelector [14] is a web mining approach focusing on structure and usage. By this approach, hyperlinks-structural relationships were extracted from existing web sites and theirs access logs. Based on the relationships, a group of hyperlinks was given to users. Using a heuristic approach, users can access the group to find the information they want.

From the related studies we overviewed above, we can see that implicit users' feedback is widely used in recommendation systems. Further, recommender systems which consist of both off-line and on-line modules have higher performance than those which only have on-line module. Moreover, concept grouping is more user-friendly because it is easier to retrieve information from a concept group than from unstructured and not interrelated pool of information.

III. FULL BAYESIAN ESTIMATION

In this study, we use Full Bayesian Estimation that has the learning function for the proposed GAM.

The proposed model analyzes the selected link of web pages, and estimates which concept class it belongs to. One link selection is one data sample. The data sample belongs to each concept class. This is expressed as in Eq. (1).

$$\text{D} = \{D_1, D_2, ..., D_n\} \tag{1}$$

where D_i (i = 1, 2, …, n) represents an aggregate of access samples of concept class that D consists of. D_1 is an aggregate of access samples of concept class D_1. D_2 , ..., D_m are the same as D_1. They are the aggregate of access samples, and belong to concept classes D_2 …, D_m respectively.

We define data sample that is used in Full Bayesian Estimation as follows. If a link that belongs to a concept class D_m is clicked, we use d_t to describe the number of click times of D_m, and d_f to describe the number of click times that concept class D_m is not clicked (i.e., other concept classes are clicked). For the history logs (not including current day), we use a variable α_t to describe the number of click times of concept class D_m, and α_f to describe the number of click times that concept class D_m is not clicked.

For example, if the whole click times is 6 at current day, and the 2 times belong to concept D_m, it means $d_t = 2$, and $d_f = 4$, then we can calculate according to Eq. (2) [15], and obtain the click probability of the concept D_m is $2/6$.

$$\theta^* = \frac{d_t}{d_t + d_f} = \frac{d_t}{d} \tag{2}$$

Equation (2) is called as Maximum Likelihood Estimation. Because the empirical value is disregarded by Maximum Likelihood Estimation, haphazardness can give a big influence on the estimation result.

But in Full Bayesian Estimation, the join of Prior Distribution (based on the history click samples) and Likelihood Estimation is used to calculate the Posterior Distribution θ. Its expression is described as follows.

$$\begin{aligned} &P(D_{m+1} = t \mid \text{Đ}) = \int P(D_{m+1} = t, \theta \mid \text{Đ}) d\theta \\ &= \int P(D_{m+1} = t \mid \theta, \text{Đ}) p(\theta \mid \text{Đ}) d\theta \\ &= \int \theta p(\theta \mid \text{Đ}) d\theta \end{aligned} \tag{3}$$

where Đ is a data collection which consists of (D_1 , D_2 , …, D_m), and is used to describe the Likelihood Estimation. The integral calculation of Full Bayesian Estimation as shown in Eq. (3) is very complicated. Generally, it needs the following premises to make it calculable.

- Each sample in Đ is independent with each other, and satisfies *iid* (independent and identically distributed) assumption;
- About the current click times d_t and d_f , theirs prior distribution satisfies Bate Distribution $B[\alpha_t, \alpha_f]$.

Thus, the Full Bayesian Estimation formula can be expressed as follows [15].

$$
\begin{aligned}
P(D_{m+1} = t \,|\, \text{Đ}) &= \int \theta p(\theta \,|\, \text{Đ}) d\theta \\
&= \frac{\Gamma(d_t + \alpha_t + d_f + \alpha_f)}{\Gamma(d_t + \alpha_t)\Gamma(d_f + \alpha_f)} \int \theta \theta^{d_t+\alpha_t-1}(1-\theta)^{d_f+\alpha_f-1} d\theta \\
&= \frac{d_t + \alpha_t}{d_t + d_f + \alpha_t + \alpha_f} \qquad (4)
\end{aligned}
$$

According to Eq. (4), if the number of the current samples is small, prior distribution has a big contribution on the result. On the contrary, if the number of the current samples is big, prior distribution has a little contribution on the result.

IV. A RECOMMENDER SYSTEM BASED ON GAM

In this study, we propose an information recommender system that is based on GAM (Gradual Adaptation Model), which consists of Concept Analyzer, Probability Estimator, and Gradual Adaption Recommender, as shown in Fig. 1.

The Concept Analyzer is used to analyze each user's access data representing his/her behaviors, and record the access data into logs. The Probability Estimator is used to estimate reuse probability of concept class for each user. The Gradual Adaption Recommender is used to analyze users' access logs and return recommendation results to gradually adapt to the transition of users' focus of interests.

The major features of the proposed system are described as follows.

- We divide users' interests into three terms of short, medium, long periods, and by remarkable, exceptional categories - which either pay a great attention to users' access behavior at current moment, or focus on casual user access.
- This system is an adaptive one. It can adapt to a transition of users' information access behaviors.
- In the system, training is not needed.

GAM is established based on Full Bayesian Estimation introduced in the previous section for estimation of user information access. This model consists of off-line and on-line components. The off-line component is used to analyze each user's access logs periodically and estimate concept classes' reuse probability for each user. The on-line component is used to analyze users' current access behavior and return recommendation results to gradually adapt to the transition of users' interests.

Fig. 1 shows the basic constitution of the proposed model (GAM), which consists of four phases, namely data pre-processing, access logs analysis, probability

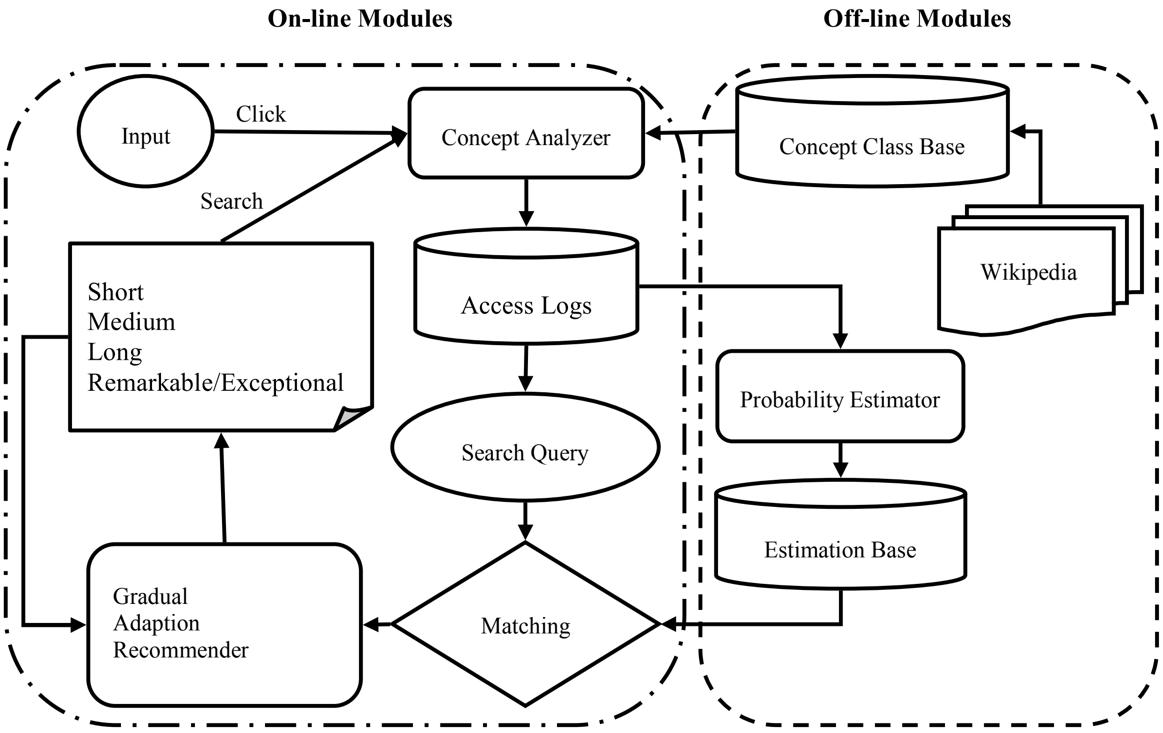
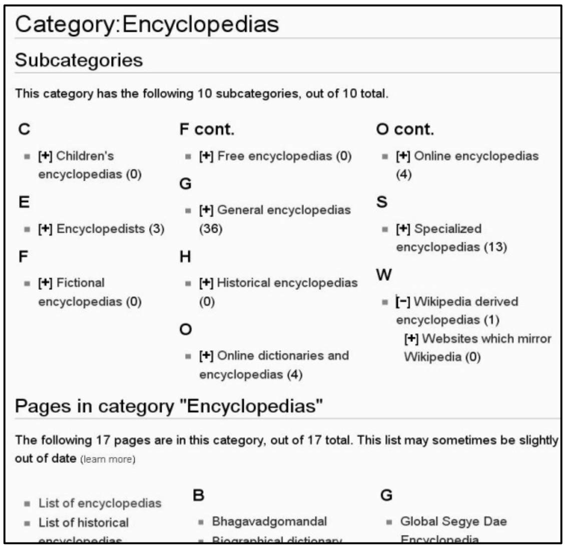


Figure 1. Gradual Adaption Model

estimating, and gradual adaptive recommendation.

A. Data Pre-processing



Category:Encyclopedias

Subcategories

This category has the following 10 subcategories, out of 10 total.

C

- [+] Children's encyclopedias (0)

E

- [+] Encyclopedists (3)

F

- [+] Fictional encyclopedias (0)

F cont.

- [+] Free encyclopedias (0)

G

- [+] General encyclopedias (36)

H

- [+] Historical encyclopedias (0)

O

- [+] Online dictionaries and encyclopedias (4)

O cont.

- [+] Online encyclopedias (4)

S

- [+] Specialized encyclopedias (13)

W

- [−] Wikipedia derived encyclopedias (1)
  [+] Websites which mirror Wikipedia (0)

Pages in category "Encyclopedias"

The following 17 pages are in this category, out of 17 total. This list may sometimes be slightly out of date (learn more)

- List of encyclopedias
- List of historical encyclopedias

B

- Bhagavadgomandal
- Biographical dictionary

G

- Global Segye Dae Encyclopedia

Figure 2. The Structure of Subcategory

At the data pre-processing phase, concept class base is created. Wikipedia [16] has 12 major categories. In each subcategory, there are pages and theirs sub-categories. Fig. 2 shows the structure of subcategory "Encyclopedia" [17] in Wikipedia. It has not only its subcategories, but also its pages.

Wikipedia category is regarded as a concept class in the proposed system. Because its pages can represent the subcategory, they are used to create index data of subcategories. Fig. 3 is the image for how to extract concept classes form Wikipedia categories.

A solid line text box means a category, or a concept class. A dotted line text box means a page, or an index data.

At first, Wikipedia categories are used to create a set of concept classes by one-to-one relationship. Then, all of pages are used to create indices for the categories that they belong to. Especially, if a category owns more than one page, its index data will be created from all of the pages as shown by the "Index 11+12" in Fig. 3.

We know that each word does not have the same importance in a page, and we need to give the weight to the words based on the importance in a page. Obviously,

- The words that are used in the title or headline of a page ought to have higher weight.
- A high frequency content-representative words are also more important for a page.

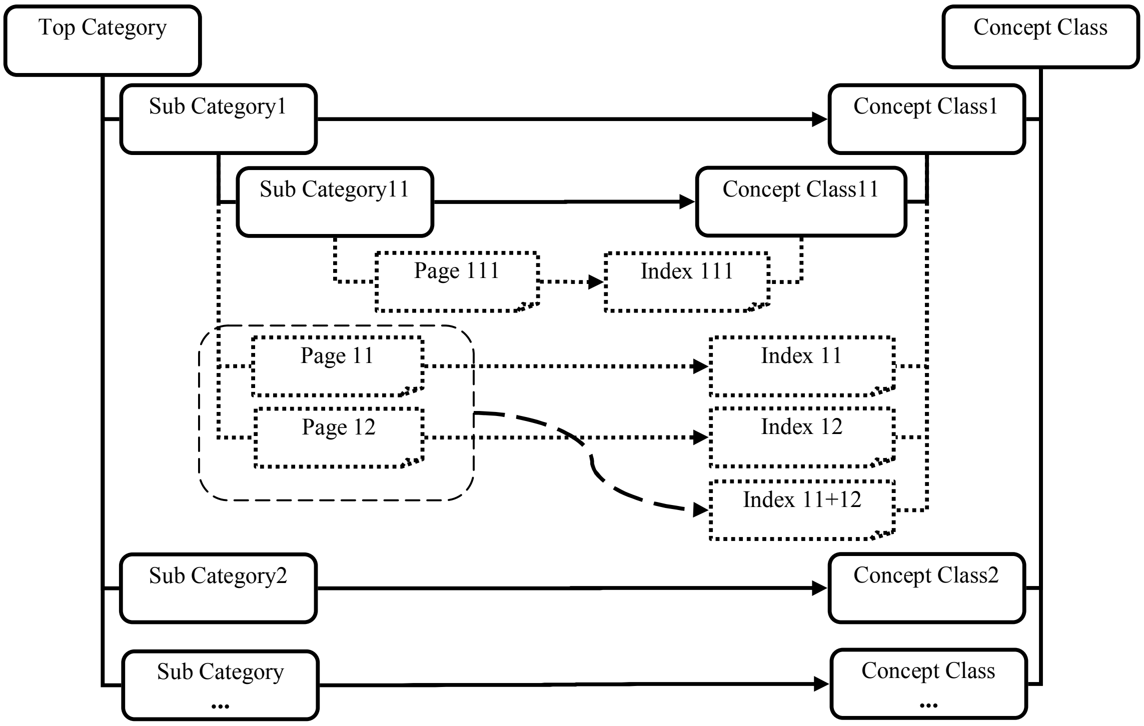


Figure 3. Concept Class and Index Extraction

- Embedded links are also given the higher weight.

Based on the above consideration, the weight is given to the items when creating indices.

Concept class extraction is a pre-processing step of the proposed system. After creation of a set of concept classes and theirs indices is done, the information recommender system can be started.

When users interact with the proposed system and provide feedback information at the first time, the system can use previously prepared index information of pages and the prior probability of concept class to give out the appropriate results to users. After users select some of results, then the access logs of user selections are used to calculate the posterior probability of concept classes. The details on how to record access logs and calculate posterior probability as discussed in Section III.

Each concept class consists of a number of keywords and URLs of web pages. Constitution of concept classes is shown in Fig. 4. It shows there are multiple concept classes in Concept Class Base. Each concept class owns multiple keywords, and some of keywords belong to multiple concept classes. When users access this system, their identifying information, clicked concept classes that include links, keywords, access date, click frequency are recorded in Access Logs. For example, a user browses recommendation results on keyword "culture" and clicks a link belonging to concept class "Art". As shown in Fig. 1, the user query will be sent to Concept Analyzer. After receiving this query, Concept Analyzer will analyze the query, and check the user's identifying information, concept class and keywords.

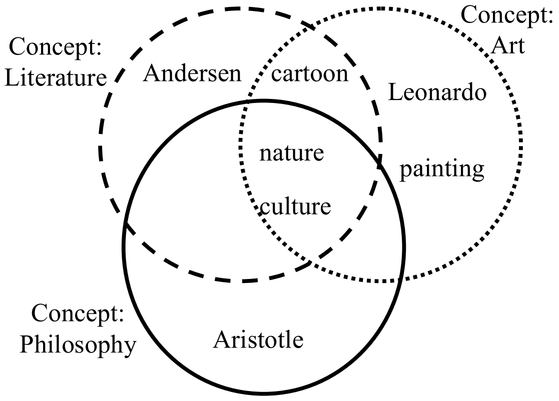


Figure 4. Constitution of Concepts

According to Fig. 4, because the clicked link belongs to concept class "Art", the sample number of concept class "Art" will be increased, and concept class, keywords, user identifying information, access date will be recorded into Access Logs. For the other concept classes (concept classes that were not accessed), there is nothing to do.

Considering the weight of keywords, Eq. (2) can be changed as follows.

$$\theta^* = \frac{\sum_{j=0} f(w_{mj}, k_{mj})}{\sum_{i=0}\sum_{j=0} f(w_{ij}, k_{ij})} \tag{5}$$

where, w_{ij} is the weight of a keyword, and k_{ij} is the selected sample number of the keyword in concept class i. Using Eq. (5), we can calculate the prior probability of each keyword from indices. As to the keyword's weight, if the keyword is in the text body, its weight is set to small values. If it is in the headline, its weight is higher than the former. If it is in an embedded link, its weight is regarded as a value between those that can be given to a headline and a text body.

B. Access Log Analyzing

When users use the proposed system, their access data representing their behaviors are analyzed and recorded by Concept Analyzer.

As the reason described above, the weight of keyword is also used to measure the selected data sample number. If the search keyword is only one, it means the sample number of this keyword increases by 1. If there are a number of search keywords, the sample number of access is divided into each keyword by its weight as follows.

$$s_m = \frac{\sum_{j=0} f(w_{mj}, k_{mj})}{\sum_{i=0}\sum_{j=0} f(w_{ij}, k_{ij})} \tag{6}$$

where i is the number of keywords, and each keyword has j weight types in the selected page. This result is recorded into the Access Logs of proposed system.

User interests are not static. They may change with time and/or environment. Therefore, we try to analyze user access logs by three periods: short, medium and long.

C. Probability Estimating

For the definition of periods, in this study, a fixed period is applied, though dynamic approaches and mechanisms as proposed in [18] can also be considered. To be simplified, in this paper, we assume the short period to 7 days (a week), the medium period to 30 days (a month), and the long period to 90 days (a quarter) (as shown in Fig. 5). All of the three periods start at previous day (-1) and do not include the current day. The short period is designed to reflect temporary interests of users. The medium period is designed for an interest that is affected by some factors, i.e., this interest is relatively stable during a period. The long period is designed for a long-term user interests. In Section V, the different features of these three periods will be shown by simulation.

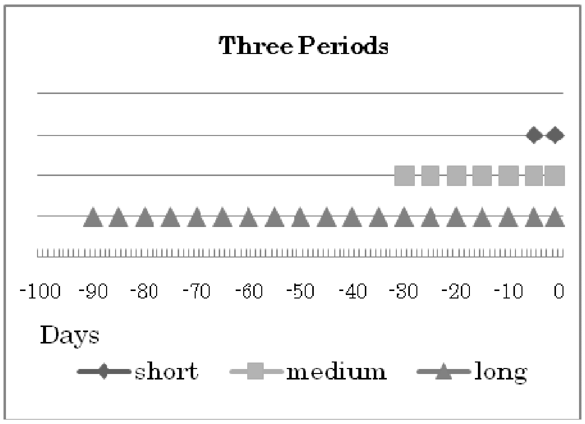


Figure 5. The Definition of Each Period

Except the three periods, we design two specific categories, namely remarkable and exceptional. Remarkable is based on the three periods. If there is a concept class belongs to more than one period, we call such concept class as remarkable concept class. The remarkable concept class means high degree of interest of a user in a particular concept class. There is also another category called exceptional. Exceptional category is an aggregate of a concept class that has a little chance to be clicked by users, but may be useful occasionally in the future for users.

The part located in the right side of Fig. 1 and surrounded by dotted line is the off-line component of system. As an off-line component, it attempts to improve the performance of the system. Probability Estimator is a part of off-line component and used to estimate the probability of concept classes. It is designed as a batch process and runs at a specific time (e.g., at the midnight) of every day.

The probability estimation is based on Eq. (4). For example, if we need to estimate a probability of concept class "Artists" and it is about user A in short period, four data items are necessary. The one pair is the sample number clicked and non-clicked by user A at the current day. The other pair is the summation of sample number clicked and non-clicked by user A in short period. Using these data, the estimator can calculate the probability of concept class "Artists" in short period. The estimator can also calculate the probabilities of other concept classes in the same way. Thus, Estimation Base can be created.

D. *Gradual Adaption Recommendation*

After creating Estimation Base, the system can start

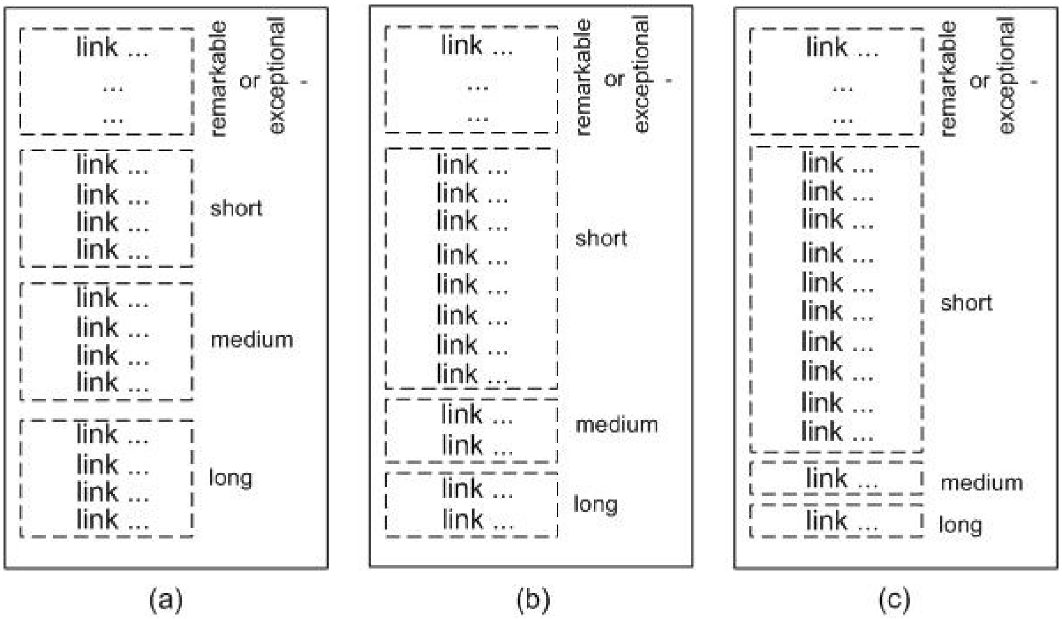


Figure 6. Three periods and two categories

the recommendation for users. The GAR (Gradual Adaption Recommender) is an on-line component. It is shown in the left part of off-line component in Fig. 1, and surrounded by dotted line.

When a user sends a search query to the system, GAR will check if there is a remarkable concept class from Estimation Base. As shown in Fig. 6, if remarkable concept class exists, GAR will return links of remarkable concept class and put them at the top of a recommendation page, choose a certain number of links from each period respectively, and add them below the remarkable links. Of course, these links belong to the concept class which has high probability in each period.

If a remarkable concept class is not found, GAR will check if an exceptional concept class exists. If an exceptional concept class exists, GAR will choose links of the concept class, then choose links from each period, and return the result in a random manner. The same as in the previous case, these links belong to the concept class which has high probability in each period.

If both remarkable and exceptional concept classes do not exist, GAR will choose the same number of links from each period respectively. These links belong to the concept class which has high probability in each of them. Then, these links are returned to a user in a random fashion.

Using the described approach, GAR gives a user a hint about which concept class is their hot concept class or which concept class is the concept class they almost forgot.

Fig. 6 (a) is the first response to a user. If the user makes a decision on a link and click it, the concept class, keyword and period or category information about the link will be sent to the system. Obtaining such information, the system will apperceive the user's demands.

As show in Fig. 6 (b), if the selected link belongs to short period, the links number of short period will be doubled. At the same way, the links number of other terms will be reduced to half.

As show in Fig. 6 (c), if the link of short period is clicked continuously, the links number of short period will be increased to a maximum number, and the number of other links of each period will be reduced to a minimum number. If a link that belongs to the short period is not clicked continuously, and another link that belongs to the other period is clicked - for instance, a link that belongs to the long period is clicked - in this case, the number of recommended links for the short period will be reduced to half, and at the same time, the number of links for the long period will be doubled.

The same things occur in case of other periods, and GAR will apperceive the change and redress the recommendation result. Therefore, GAR can give a high satisfaction rating to users.

V. SIMULATION AND EVALUATION

In order to verify the operability of the proposed GAM, we pre-produced the model. The system was built by open source software: Java, Tomcat, MySQL, and Nekohtml were used. Using them, Concept Analyzer, Probability Estimator, Gradual Adaptive Recommender were built

For the simulation, we consider three basic cases to evaluate the system. The first case is a user who has a long-term interest. In this case, the probability of the interested concept class ought to keep a high rate in long period.

The second case is a user who has a temporary interest. The user access the concept class of temporary interest sometime. In this case, this concept class ought to keep a low rate in the three periods.

The third case is a user who has two interests, and these interests are affected by some factors easily. In the case, there is a possibility that the probability of the relation concept class can change hugely in the short or medium period, but not in the long period.

A. Concept Class Base

We use Wikipedia on DVD Version 0.5 [19] (we refer to it as Wikipedia 0.5 for brevity) as the test data. The lowest categories in Wikipedia 0.5's topic hierarchies are used as the concept classes in the simulation. Based on Wikipedia 0.5, we gained more than 2000 web pages that belong to 180 concept classes. These concept classes were ready in advance, and saved in the Concept Class Base.

B. Setting of Simulation Cases

For case one, the concept class "Philosophical thought movements" is assumed to be used every day, and the assumed number of clicks (a user's accesses) was set to 0 and 40 per day. It means the user is interested in the concept class, and has a long-term interest in the concept class.

For case two, the concept class "Philosophers" is assumed to be used per three days, and the assumed number of clicks was set to 0 and 10. It means the user has little interest in this concept class.

For case three, two concept classes of "Art" and "Artists" are assumed to be used, and the number of clicks is dynamically varying. Most times it is set to 0 to 20, but sometimes it is set to 0 to 10 (likes case two), some other times it is set to 0 to 80 (large than case one).

Obviously, concept classes "Philosophers" and "Philosophical thought movements", and concept classes "Art" and "Artists" are similar respectively in cases described above. It means that similar concept classes have similar keywords. We expect that our model can differentiate the concept classes even if they contain similar keywords, and gain the results as we explained above.

C. *Simulation Results*

We simulated the three test cases during a period of 150 days, and obtained the results. The results are what we expected, showing high adjustability and adaptability of the proposed model.

In the short period, we can see the movement of the concept rate changing frequently. In some days, the probability of concept classes in case three is bigger than case one – for instance, "Art" concept class gets higher probability at 2008/12/12 point , "Artists" concept class gets higher probability at 2008/10/31 point (Fig 7).

In the medium period, the change is becoming smaller. But the probability of concept classes in case two is bigger than case three in some days (Fig 8). The exchange of probability between "Art" and "Artists" is also can be seen at 2008/10/31 and 2008/11/28 point.

In the long period, the change becomes quite stable. There is no big change in the long period (Fig 9).

From the simulation results, we found that the proposed model adapts well to the change of user's interests, as we expected. Thus, if a concept class is used frequently, it ought to have a high probability in the long period. If the concept class is used to a certain extent, it ought to have a quick change in the short or the medium period. If the concept class is used rarely, the rate ought to keep at a low level. This result demonstrates that the proposed model is operable and effective for modeling situations similar to those in the above-mentioned cases.

VI. CONCLUSION

In this study, we have proposed a gradual adaption model (GAM) for estimation of user information access behavior, based on Full Bayesian Estimation with a learning function, in order to solve the uncertainty problem caused by differences in user information access behaviors. A variety of users' information access data are collected and analyzed in terms of short, medium, long periods, and by remarkable and exceptional categories. We have further implemented a prototype system based on the proposed model, designed experimental simulations with three assumed cases to show operability and effectiveness of the model.

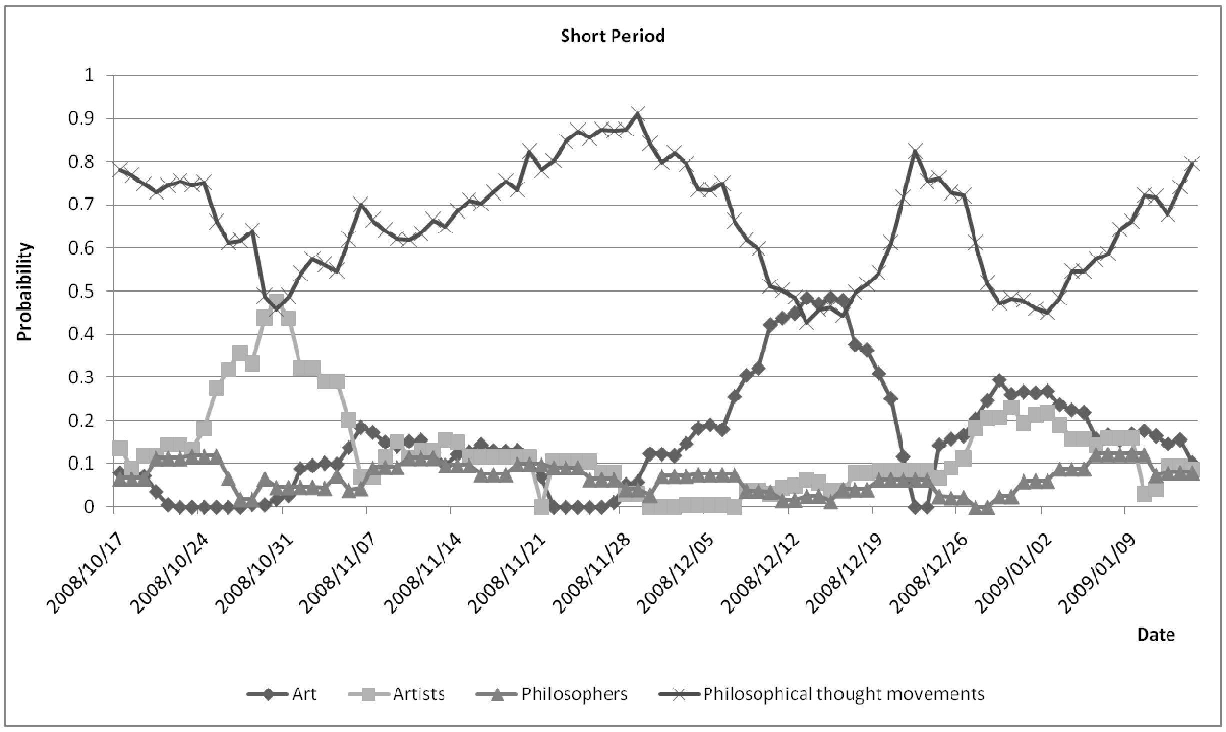


Figure 7. Probability of Concept classes in Short Period

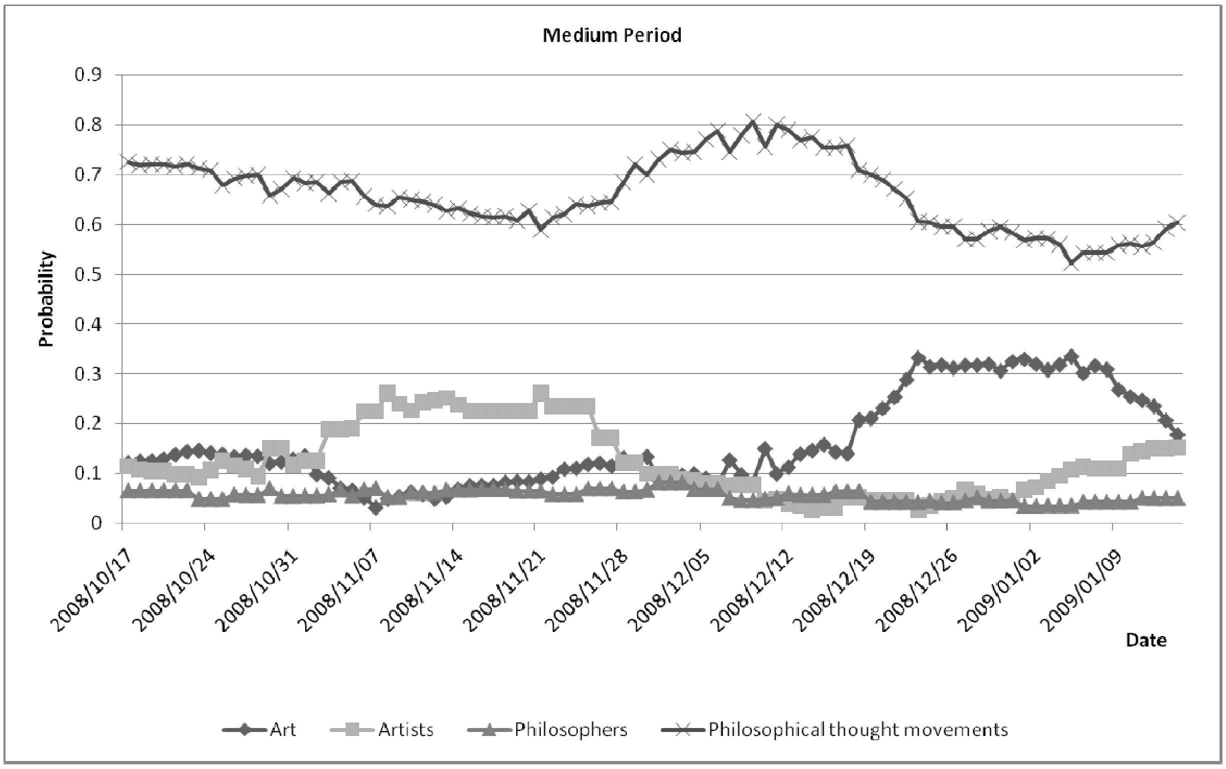


Figure 8. Probability of Concept classes in Medium Period

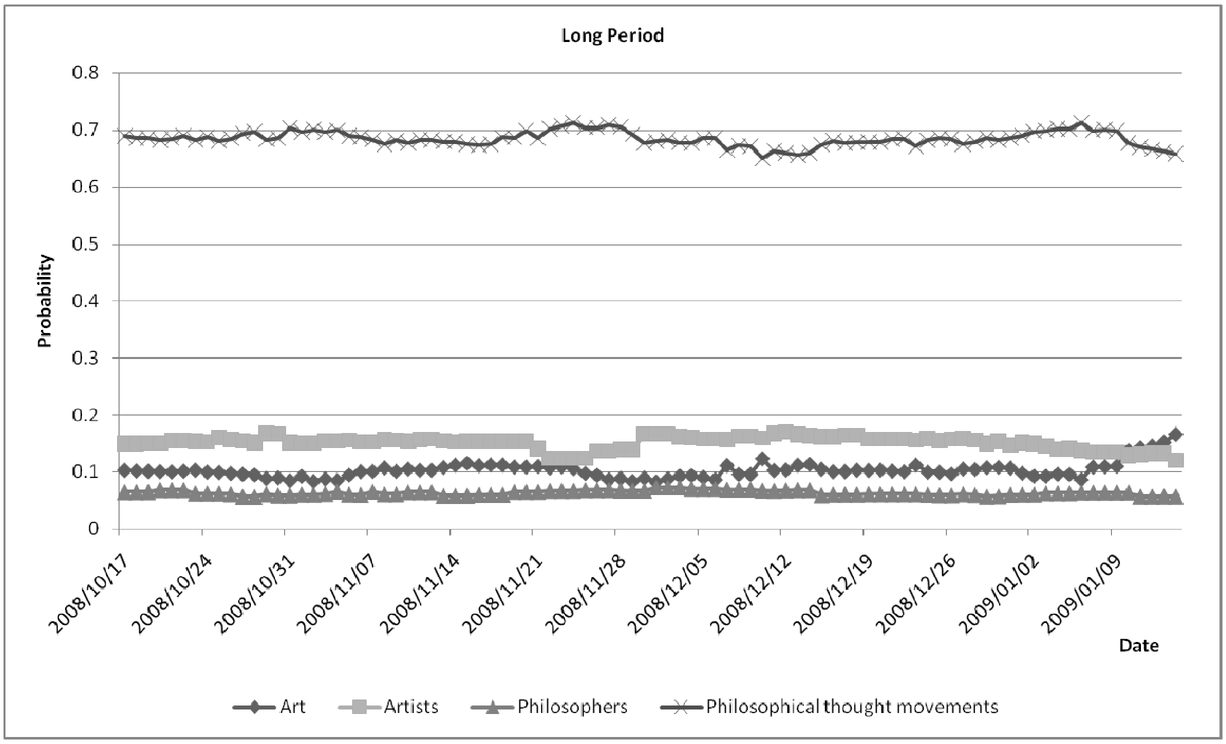


Figure 9. Probability of Concept classes in Long Period

The simulation results have shown that the proposed model can recognize the transition of users' access behaviors (web page selections, in particular) sensitively in the short period. The users' long-term interest is kept a high probability in the long period. The three periods of GAM can correctly distinguish long-term and temporary interest of users. Based on the results, when a user inputs a keyword and selects a link of a concept class that belongs to the long period, GAM can return the links of the concept classes that belong to the long period and match with the input keyword. Because the other concept classes that belong to the short and medium periods are filtered, GAM can help user to find the information that he/she is seeking quickly. Of course, GAM can detect which period is focused by a user, therefore, it can gradually adapt to the transition of users' selection, and provide appropriate information to the user.

As for future works, we will set more different patterns for the short, medium and long periods to find more reasonable ones. Using a dynamic sampling to set the three periods is one of the future works. Moreover, we will implement a fully runnable system, and evaluate the proposed model with users' involvement. We expect such experiment results can give us insights on how to further improve the model. We will also compare the proposed approach with other related recommendation models.

REFERENCES

[1] J. Chen, R. Shtykh, Q. Jin, "Gradual Adation Model for Estimation of User Information Access Behavior," ICSNC '08, pp. 378-383.

[2] M. Kashihana, S. Takeshi, Y. Endo, R. Doi, "Evaluation of Wikipedia (in Japanese)," March 2008.

[3] D. Milne, O. Medelyan, Ian H. Witten, "Mining Domain-Specific Thesauri from Wikipedia: A case study," Proc. IEEE/WIC/ACM International Conference on Web Intelligence (WI' 06), Hong Kong, China, 2006, pp. 442-448.

[4] R. Mihalcea, A. Csomai, "Wikify! Linking Documents to Encyclopedic Knowledge," CIKM'07, Lisboa, Portugal, November 2007, pp. 233-241.

[5] Y. Wang, H. Wang, H. Zhu, Y. Yu, "Natural Language Processing and Information Systems," Springer Berlin / Heidelberg, August 2007, Vol. Volume 4592/2007.

[6] C. Thomas, Amit P. Sheth, "Semantic Convergence of Wikipedia Articles," IEEE/WIC/ACM International Conference on Web Intelligence (WI'07), Silicon Valley, USA, 2007, pp. 600-606.

[7] J. Srivastava, R. Cooley, M. Deshpande, P.-N. Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data," ACM SIGKDD, Vol 1, Issue 2, Jan. 2000, pp. 12–23.

[8] B. Poblete, R. Baeza-Yates, "Query-Sets: Using Implicit Feedback and Query Patterns to Organize Web Documents," Proc. WWW2008, Beijing, China, Apr. 2008, pp. 41-48.

[9] G. Stumme, A. Hotho, B. Berendt, "Semantic Web Mining State of the Art and Future Directions," Elsevier Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 4, No. 2, 2006, pp. 124-143.

[10] M. Bilenko, R. W. White, "Mining the Search Trails of Surfing Crowds: Identifying Relevant Websites From User Activity," Proc. WWW2008, Beijing, China, Apr. 2008, pp. 51-60.

[11] M. Eirinaki, M. Vazirgiannis, "Web Mining for Web Personalization," ACM Transactions on Internet Technology, Vol. 3, No. 1, 2003, pp. 1–27.

[12] T-W. Yan, M. Jacobsen, H. Garcia-Molina, U. Dayal, "From User Access Patterns to Dynamic Hypertext Linking," Proc. WWW1996, Paris, France, May 1996, pp. 1007-1014.

[13] R. Baraglia, F. Silvestri, "An Online Recommender System for Large Web Sites," Proc. IEEE/WIC/ACM International Conference on Web Intelligence (WI'04), Beijing, China, Sep. 2004, pp. 199-205.

[14] X. Fang, O.R. Liu Sheng, "LinkSelector: A Web Mining Approach to Hyperlink Selection for Web Portals," ACM Transactions on Internet Technology, Vol. 4, No. 2, May 2004, pp. 209–237.

[15] L. Zhang, H. Guo, *Introduction to Bayesian Networks (in Chinese)*, Science Press, 2006.

[16] http://en.wikipedia.org/wiki/Portal:Contents/Categorical_index

[17] http://en.wikipedia.org/wiki/Category:EncyclopediEn

[18] J. Chen, R. Shtykh, Q. Jin, "A Web Recommender System Based on Dynamic Sampling of User Information Access Behaviors," submitted to CIT'09, Xiamen, China, Oct. 2009.

[19] http://www.wikipediaondvd.com/nav/art/d/w.html

Resources Sharing and Access Control in Group-oriented Networks: Fednet and Related Paradigms

Malohat Ibrohimovna
Technical University of Delft
The Netherlands

K.M.Ibrohimovna@ewi.tudelft.nl

Sonia Heemstra de Groot
Twente Institute for Wireless and Mobile Communications and
Technical University of Delft
The Netherlands

sonia.heemstra.de.groot@ti-wmc.nl

Abstract

A Personal Network (PN) is a network composed of devices of a person that can communicate with each other independently from their geographical location. Extra functionality in PNs enables the cooperation amongst different persons forming a group-oriented network called a Federation of Personal Networks (Fednet). A Fednet is a secure, opportunity or purpose driven ad-hoc network for sharing personal resources. A Fednet can be composed for applications in different areas, e.g. education, entertainment, business, emergency, etc.

A number of group-oriented resource-sharing technologies for distributed environments have been reported in the literature, such as grids, Virtual Organizations, Secure Virtual Enclaves and P2P networks. All these technologies for sharing resources have their own peculiarity in the architecture, their implementation, and in the ways they control the access to shared resources. This paper provides a comparative overview of these technologies with our Fednet concept. In addition, a special attention is given to various approaches for controlling the access to shared resources in cooperative distributed environments, in particular grid environments. We discuss the details of these access control architectures, advantages and disadvantages of these approaches.

Keywords: sharing resources, group-oriented networks, personal networks, federation of personal networks, access control.

1. INTRODUCTION

Personal devices with networking capabilities have become an integral part of daily activities, business and entertainment. Examples of such personal devices are mobile phones, PDAs, digital cameras, laptops, desktops, MP3 players, printers, home appliances, gadgets, etc. It is exciting and useful, when these personal devices and appliances could communicate with each other and provide meaningful services to their owners independently of their geographic location. This is the idea behind the concept of a Personal Network (PN) [2].

The personal devices in a PN are organized into clusters. A cluster is a networked group of personal devices located in the vicinity of each other. A simple PN consists of a local cluster around the user. Figure 1 illustrates an example of a PN. In this PN a local cluster is extended with other remote clusters, i.e. office cluster, home cluster and car cluster with the help of interconnecting infrastructures. This way, personal devices can form a distributed personal environment of a user.

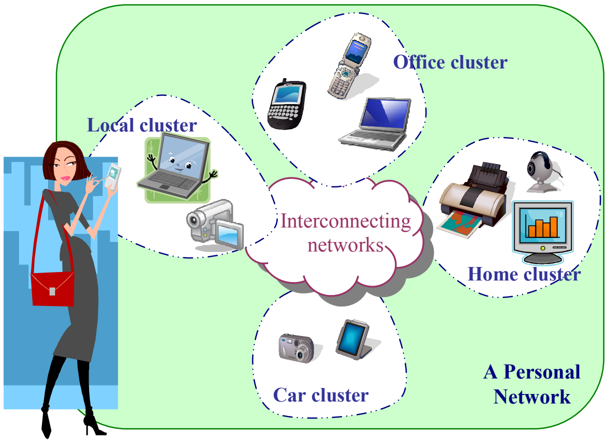


Figure 1. Example of a Personal Network

By adding extra functionality, PNs can form group-oriented networks called Fednets. The concept of a Fednet was introduced in [3] and defined as a temporal, ad hoc, opportunity- or purpose-driven secure cooperation of independent PNs. In the core of the Fednets is sharing personal resources and services to achieve a common objective.

A number of technologies and paradigms for sharing resources have been reported in the literature. But in each of them the concept of 'sharing' appears with a new flavor: in grid computing [4] it is sharing the spare CPU resources, processing power and storage facilities; in P2P networks it is sharing data and multimedia content, such as music, clips and video; in Wireless community networks [5] it is sharing services and facilities such as Internet access; and in Fednets it is a broad range of sharing personal resources and services among the users on demand. All these technologies for sharing resources differ in their architecture, their implementation, and in the ways they control the access to shared resources.

The contribution of this paper is a comparative overview of several group-oriented resource-sharing technologies for distributed environments. In addition, a special attention is given to various approaches for controlling the access to shared resources in cooperative distributed environments, in particular grid environments. We discuss the details of these access control architectures, advantages and disadvantages of these approaches. In this sense, this paper extends the survey on resource sharing technologies presented in [1] (UBICOMM 2008).

The organization of the paper is the following. In Section 2, we explain the motivation for PNs to federate, and briefly describe the basic component-level architecture of a Fednet and access control in Fednets. Further in this section, we explain our approach to analyze the system based on functional modules. In Section 3, we discuss some of the related technologies for group-oriented communication reported in the literature. In Section 4, we analyze the access control mechanisms used in grid environments based on the generic authorization framework for Internet resources and services [24]. Finally, in Section 5, we summarize the survey and draw conclusions.

2. FEDNETS

In this section, we describe Fednets, present their architecture and the access control to its resources.

2.1 Motivation to federate PNs

Persons usually communicate with each other, carry out common tasks and cooperate with each other in order to reach a common goal. They might encounter many situations, when it is desirable and beneficial to enrich their cooperation by connecting their Personal networks for raising the efficacy of their communication towards reaching a common goal. A network that is created by connecting independent PNs is called a Federation of Personal networks (Fednet). The concept of a Fednet was introduced in [2] as a temporal, ad hoc, secure cooperation of independent PNs. PNs, driven by a certain purpose or triggered by opportunity, can form a Fednet to achieve a common objective by means of sharing personal resources and services. Figure 2 depicts an example of a Fednet formed of four PNs to share their resources and services. In this Fednet the PN owners can run different applications and benefit from sharing personal resources (e.g. data and multimedia) and personal services (e.g. printing, displaying, storing, connectivity to the infrastructure, routing and Internet access).

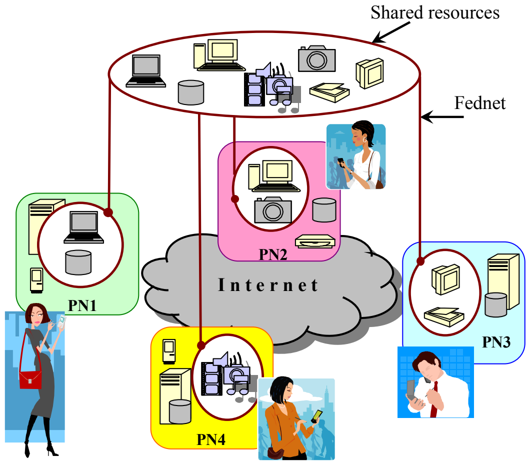


Figure 2. A Fednet and its shared resources

The main objective in Fednets is to facilitate reaching a common task. For example, consider the following scenarios:

a) *File sharing*. Colleagues attend a conference together. They meet after the conference and form a Fednet with the goal to share photos and videos from the conference. The Fednet here is purpose driven and is formed between laptops, photo cameras and PDAs of the colleagues. This example is elaborated in [6].

b) *Camera view and sensor information sharing*. People with wearable cameras and sensors can form a Fednet with the goal to exchange valuable information (e.g. images, temperature and location) in disaster relief situation. The Fednet here is purpose driven and is composed by wearable cameras and sensors of different people.

c) *Facility sharing*. Friends meet at home of one of them. They show each other photos in their iPods. The host has a big screen. Using this opportunity, they form a Fednet with the goal to display the pictures on a big screen. The Fednet here is opportunity-driven and is formed between the iPods of friends and the screen.

These examples show the wide applicability of Fednets in various situations for ad hoc occasional sharing of personal resources. It is important to note, that Fednets can have a large scale involving a large number of distributed

PNs. Examples of such Fednets are a secure network between patients and doctors for remote healthcare services or a virtual classroom environment formed for a distance learning course scenario.

2.2 Architecture of Fednets

A Fednet is composed of interconnected PNs. PNs belong to different owners and represent independent security domains; therefore the architecture of a Fednet should take into account the following considerations:

• The resource and service owners might want to keep the control over their resources and services themselves;

• The internal structure of a PN is not to be revealed to other PNs.

Two approaches [7], [8] have been taken so far to build the architecture of the Fednets: using overlays between PNs [7] and using service proxies at the gateways of the PNs [8]. The difference between the two approaches is in the way service access control and service provisioning are carried out. In the overlay approach, each personal device in the Fednet carries out the service access control, while providing a service to others.

In the proxy-based approach, the services of a PN are accessed by other PNs not directly at the service providing device, but at the gateway of a PN by means of service proxies. Besides, the access control to the PN services is carried out not in every device in a PN, but at the border of the PN (i.e. the gateway of a PN), so other personal devices inside the PN do not need to have access control capabilities. Having the access control at the borders, allows each PN to have a separate security domain in a Fednet and to keep its autonomy. This way, the proxy-based approach meets better the above-mentioned considerations. This advantage in comparison to the overlay-based approach has been our main motivation for choosing for the proxy-based architecture in our work. Figure 3 illustrates the basic proxy-based architecture of a Fednet.

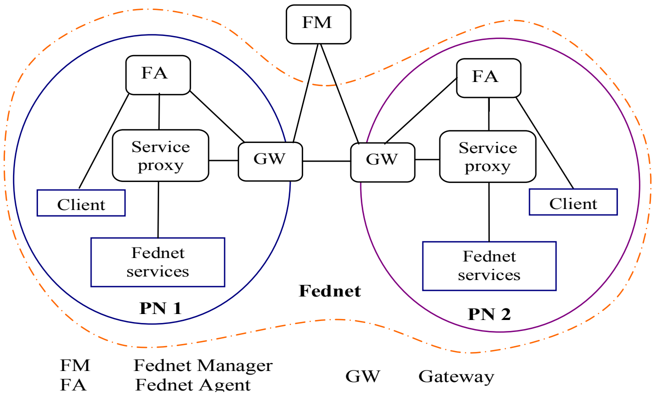


Figure 3. Basic proxy-based architecture of a Fednet

The main components of a Fednet are the Fednet manager (FM) and the Fednet agent (FA). The Fednet manager is responsible for management and control of the Fednet, such as creating and dissolving a Fednet, and accepting and removing Fednet members. The Fednet agent is responsible for the management and control functions of the PN when operating within a Fednet, such as joining and leaving a Fednet, controlling access to its personal resources and services. The Gateway (GW) is a device with multiple network interfaces. A PN communicates with other PNs of the Fednet through this gateway, by making one of its interfaces publicly addressable. The Service Proxy is a functional component that is located at the gateway of the PN. Its role is to prevent direct access of other Fednet members to the personal devices (services) of a PN, by making the services available at the gateway of a PN. The services offered by the PN to the Fednet are called Fednet services. A client is an application or a personal device within a PN requesting a Fednet service. The proxy-based architecture of a Fednet is given in more details in [6].

2.3 Access control in Fednets

In Fednets two or more PNs share their resources and services with each other. When people share their personal resources and services, an important issue is providing a proper access control to them. We took a two-level approach for the access control in Fednets. We consider that *becoming a member* of the Fednet (i.e. access to the community) and *using the services* and resources of the Fednet (i.e. access to the community services and resources) are two different issues. Therefore, we distinguish between these two levels of the access control. The two-level approach gives a separation of concerns in the access control.

The first-level access control takes place when a new member joins a Fednet. The first-level access control in a Fednet is carried out by the Fednet manager. Having a centralized entity (i.e. the Fednet Manager) for this task facilitates the management of the Fednet with dynamically joining and leaving members. The accepted Fednet members receive a membership credential which is used to prove their membership within the Fednet. Membership credential also indicates a PN's *membership class*, which is the ranking of the member within the Fednet based on its contributions and reputation.

The second-level access control is the access control to the Fednet services. It takes place when a Fednet member requests a Fednet service. The second-level access control in a Fednet is carried out by the Fednet agent of a PN. This allows a PN to keep the control over its personal resources and services. This meets the preferences of the PN owners, who usually prefer not to delegate the access control rights over their personal resources to a third party.

Fednet services are distinguished between common and specific. Common services are accessible by all members of the Fednet upon presenting their membership credential. Special services require the second-level fine-grained access control at the PNs.

2.4 Functional modules in the architecture of Fednets

In this paper, we focus on the *concept of sharing resources*, in particular on the following questions:

- Who is sharing?
- What is shared?
- How is the sharing done?

To analyze the system based on these questions we introduce a system architecture decomposed into functional modules, as is shown in Figure 4. By *module* we mean a collection of functional components.

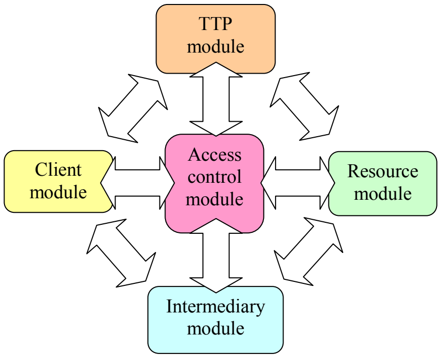


Figure 4. Functional modules of system architecture.

The following functional modules are relevant when sharing resources between different owners:

• *Client module,* which contains the administrative domains and the users of shared resources.

• *Trusted Third Party* (TTP) module that contains a trusted authority between all administrative domains.

• *Intermediary module* that contains technology-specific components of the architecture.

• *Access control module* that contains the mechanisms or methods used in the access control to shared resources.

• *Resource module* that contains types of shared resources and services.

We took this approach to analyze the system, because it gives us better understanding of how the system works and how the sharing is accomplished. It shows explicitly the interrelation of functionalities in the sharing process. It also helps us to compare Fednets and other related technologies with respect to the concept of sharing resources.

Figure 5 shows the mapping of the Fednet architecture, depicted in Figure 3 into functional modules grouped according to the above mentioned criterion.

• In Fednets *the client module* consists of PNs that belong to different owners.

• *The TTP module* can contain a Certification Authority (CA), who issues digital certificates for PNs to certify their identity in the authentication process (see Figure 5, arrow 1).

• *The intermediary module* contains a service directory (SD), which stores the list of Fednet services; a gateway (GW) through which all external communication of a PN takes place and a Service proxy, which makes a service available at the gateway of a PN.

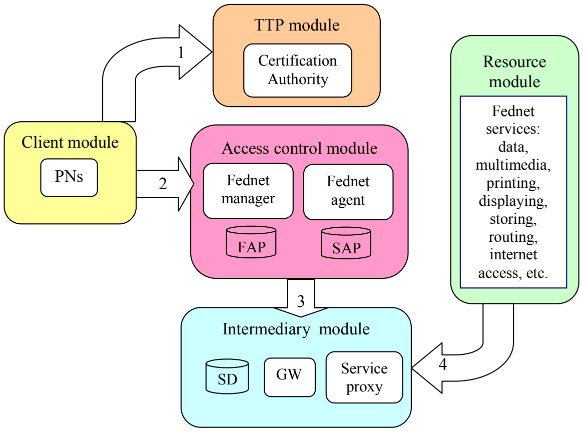


Figure 5. Functional modules in the Fednet architecture.

• *The access control module* contains the Fednet manager; Fednet access control policies (FAP), which are the rules about how a new member is accepted to a Fednet; the Fednet agent and service access control policies (SAP), which are the rules about how the access to the PN services is controlled.

• *The resource module* contains Fednet resources and services that are shared between the PNs. The access to the Fednet and its services is achieved through the access control module (Figure 5, arrow 2), which produces the access control decisions that are enforced by the intermediary module (arrow 3), where a service proxy located at the gateway of the PN acts as a delegate of a PN service (arrow 4).

In Section 3 we describe some of the related paradigms proposed in literature. In order to analyze them, we use our approach of decomposition of the system architecture into functional modules, presented in Figure 4.

3. RELATED PARADIGMS FOR SHARING RESOURCES

The main purpose of Fednets is sharing the resources and services which belong to different persons. A number of technologies and paradigms for sharing resources and services have been proposed in the literature [3], [9-11], [19-23]. While some of them focus on sharing a particular service, other systems are designed for a group of various applications.

To place the Fednets amongst related technologies we discuss some of them in this Section. We provide the definition of a technology, discuss its differences and similarities with Fednets, and analyze the functional architecture focusing on the access control to shared resources in each technology.

3.1 Grids

A grid [3] is a hardware and software infrastructure that allows resources to be shared across organizational boundaries. Grid computing was started with the idea of sharing spare processing power and storage facilities to carry out big scale computations that were not possible by using single machines. Later, organizations using grid networking started cooperation based on mutually agreed rules to form so-called Virtual Organizations (VO) [9]. There are many grid projects all over the world, such as the project CrossGrid [10], which addresses realistic problems in medicine, environmental protection, flood prediction, and physics analysis; the project AccessGrid [11], which enables connecting people using remote video, visualization techniques, microphones and cameras. An impressive amount of examples of grid projects and applications is given in [12] and [13].

Functional architecture of grids

Figure 6 depicts the functional architecture of a grid network.

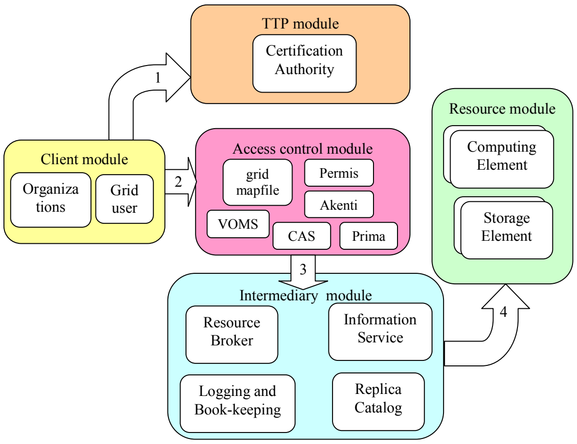


Figure 6. Functional modules in Grid architecture.

The *client module* contains the grid users, and organizations, which are the members of a VO.

The *TTP module* contains a Certification Authority (CA), who issues Grid digital certificates after certifying user's identity, for example showing staff ID.

The *resource module* contains computing and storage elements.

The *intermediary module* is responsible for managing the job allocation and execution. Grid computation shares resources online through the Internet, so anyone may access shared resources. In order to use a grid facility, the grid user first gets a certificate from the CA (see Figure 6, arrow 1) and submits a job (arrow 2) to the grid facility via the user interface. The application control mechanisms carry out the access control by checking the grid map file. This file holds a mapping list of the authenticated grid users to their local account names. When the user is found in the grid map file, the resource broker uses information service and replica catalog (arrow 3) to find a suitable computing element and a storage element to execute the job (arrow 4). When the job is done, the resource broker returns the result to the user. Logging and book-keeping service maintains the records on the job execution procedure, which are purged when the job is completed.

The *access control module* of grid networks contains several mechanisms, which are shown in Figure 6. They are grid map files, Community Authorization Service (CAS) [14], Virtual Organization Membership Service (VOMS) [15], PERMIS [16], AKENTI [17] and PRIMA [18].

The access control in grids

Grid computing provides not unrestricted sharing, but controlled sharing of resources. Resource owners typically put restrictions on the access to their resources based on the membership, payability, etc. The basic idea of controlling access to shared resources is through authentication. The simplest authentication design is to set up a username and password for the user to join a grid and to keep this information in a *grid map file*. The username is verified with a digital certificate issued by the CA. The drawback of using grid map files is that it is difficult to maintain them for a large number of grid users. More sophisticated mechanisms developed for the access control in grid environments are: CAS, VOMS, PERMIS, AKENTI and PRIMA. Section 4 provides detailed discussion on these mechanisms.

Similarities and differences with Fednets

Although the idea of sharing in grids and VOs is similar to the idea behind Fednets, there are major differences between them. First of all, the administrative domains in Fednets are personal networks, with personal resources and services. It is an overlay network between personal networks of individuals. In grids, the administrative domains are organizations and therefore it is an overlay network formed between organizations. Second, personal resources are mostly portable and battery-powered, while in grids the resources are big scale computing and storage facilities. Third, Fednets are formed on demand for temporal situations. On the contrary grid applications and projects are set up on a long-term basis to solve complex problems with long-term goals. Forth, Fednets have a dynamic nature, i.e. its constitution dynamically changes over the time, while grid networks have a static nature, with static constituent parts. Fifth, the applications of a Fednet have relatively smaller scope in comparison with grids and VOs. For example, Fednet can be formed for the Internet access sharing, file sharing, printing, display, storage, games, and entertainment, while grids and VOs are formed to solve country-wide or international problems, such as weather forecasting, air pollutions, human genome studies, etc. And sixth, is the way how the management is done. In VOs there is one (or more) professional system administrators to manage the VO, while in Fednets, the user is not a professional and should preferably not be bothered with any management/configuration task.

3.2 Secure Virtual Enclaves

SVE [19] is a middleware infrastructure that allows multiple organizations to share their distributed application objects respecting organizational autonomy. The goal of the SVE is to provide a restricted access to the resources and information databases of organizations. Controlled collaborative computing in SVE is based on using open networks and distributed application technologies, such as WWW, CORBA, Java, Active X and combinations with legacy applications.

The SVE was meant to be used in collaborative computing scenarios, such as:

• In military environments, joint task forces might share selected information and applications for distributed collaborative planning.

• In disaster or incident response teams, various government organizations and corporate units rapidly form a team. They share information in a limited way that is beyond sharing in ordinary settings.

• In business environments, corporate units share information with outside organizations without allowing general access to sensitive corporate data, only allowing authenticated controlled access to a subset of data.

Functional architecture of Secure Virtual Enclaves

An *enclave* is a set of resources (computers and networks) of an organization, which belongs to the same security domain. One or more enclaves form a SVE by joining with a subset of their resources. SVE identifies a distributed collection of selected resources, along with the principals that are authorized to access those resources. Principals are the persons, servers or programs. Figure 7 illustrates a functional architecture of SVE.

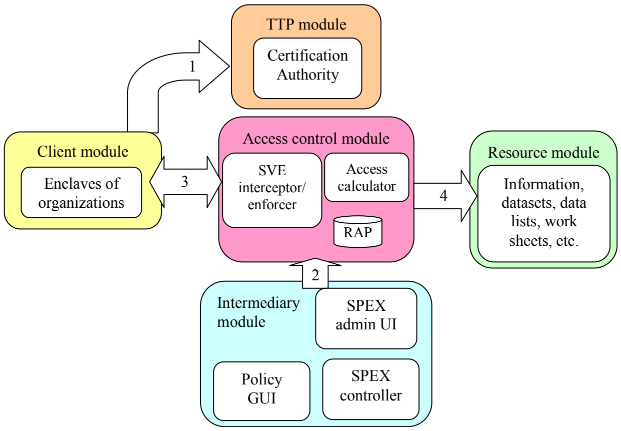


Figure 7. Functional modules in SVE architecture

The *client module* contains the enclaves of different organizations, which are the members of SVE.

The *TTP module* contains a Certification authority that issues X.509 certificates for enclaves.

The *resource module* contains application objects of enclaves, e.g. data lists, work sheets, corporate data and information.

The *intermediary module* is responsible in management and operation of SVE. It contains SVE Policy EXchange administration graphical user interface (SPEX GUI), SPEX controller and Policy GUI. The management of an enclave is carried out by the enclave administrator who administers SVE operation, e.g. initiates joining, leaving and creating the SVE. For this task, the administrator gives commands to the SPEX controller through SPEX Administration GUI. The Enclave can join several SVEs. The administrator of the enclave is also responsible for defining and maintaining the access policies for local enclave resources via the Policy GUI. SPEX controller in its turn propagates these policies within the local enclave to the SVE policy enforcement components (i.e. SVE interceptor/enforcer) and to other SVE member enclaves.

The *access control module* contains SVE interceptor/enforcer, Access Calculator and resource access policies (RAP).

Access control in Secure Virtual Enclaves

While sharing it is important for enclaves to keep autonomy, so that the control over the resources is kept in each enclave locally. Access control policies are not propagated among enclaves. The enclave has the full control over its resource access policy. Therefore, the access control to the resources is done within each enclave locally, while the administration and maintenance within the SVE is done by all enclaves together.

Enclaves authenticate each other using certificates of a Certification authority (Figure 7, arrow 1). Each local administrator determines the access to the local resources granted to the community by defining the enclave's resource access control policies (arrow 2). A request of the user from another enclave is received at the SVE interceptor (arrow 3), which queries a local Access Calculator for an access decision. Then the Access Calculator evaluates the resource access policies to grant or deny the access to the resources of the enclave. The SVE enforcer then enforces the decision by either allowing the request to proceed as usual (arrow 4), or dropping the request and returning an error message to the client. The access rights are derived by the Access Calculator in four steps: domain derivation, type definition, access matrix check and constraint check. The SPEX controller provides asynchronous policy updates to local access calculators.

In SVE access authorization is role-based and the access is granted equally to all local and foreign principals, which are represented by a domain. For example, if an individual acts in the SVE as an engineer, then he belongs to an engineer's domain. He has the rights assigned to this role to access the SVE resources regardless of his location and the location of the resources he is accessing. The autonomy of the enclave is provided by having its local policies to its resources and having the opportunity to withdraw any resources any time from the SVE. If any of the collaboration partners is found untrustworthy, the enclave can immediately modify its local policy components and update its Access Calculators.

Similarities and differences with Fednets

We can see the following differences among Fednets and SVE. First, the administrative domains in SVE are organizations. Second, the resources in SVE are application objects, the information and datasets e.g. worksheets, data lists that belong to distributed applications such as WWW, CORBA, Java, Active X. Third, the SVE applications are set up for a relatively long time to support inter-organizational activities, e.g. working on the common data lists. Forth, the applications of SVE have a bigger scope, they are meant for inter-organizational communication, while Fednets are meant for inter-personal communication. The idea behind Fednets is to enable a broad range of sharing personal services among the users on demand.

In addition, there are differences in implementation. Interceptor and enforcer in SVE are functionalities that act as intermediaries between external clients and internal servers of the organization. They are implemented at the enclave gateways or as server modifications. In Fednets, these functionalities are implemented as Fednet agents and service proxies at the gateways of the PNs. Furthermore, in Secure Virtual Enclaves (SVE), the SVE administrator defines the resource access policies of the enclave, initiates joining, leaving and creating the SVE. Its functionality is comparable with the PN owner, with the difference that the PN owner manages his/her own resources, while the SVE administrator manages the resources of the enclave that belongs to one organization. Moreover, in SVE a new enclave can join the SVE through voting if only the majority of the enclaves agree on that. Each enclave maintains the list of trusted collaborators, i.e. enclaves of other organizations. Consequently, no anonymity is supported in SVE. Fednets, on the contrary, depending on its goal and the type of its applications can have also anonymous nature, in which the members do not need to know who is in the Fednet.

3.3 Peer-to-peer file-sharing networks

A P2P network is a collection of distributed computers where each computer is called 'a peer' and shares resources and services with other peers. Peers have equal responsibilities and capabilities in providing/consuming the services.

The examples of most popular P2P applications are Napster, Gnutella, Fasttrack, Morpheus, Freenet and Kazaa.

The functional architecture

Figure 8 shows the functional architecture for a typical P2P file-sharing network.

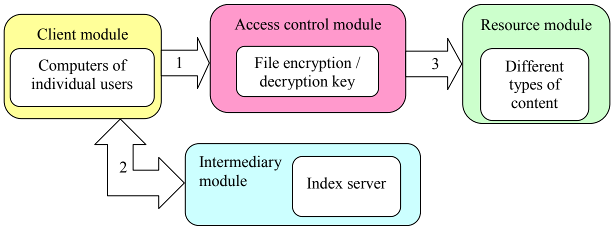

Client module
Computers of individual users
1
Access control module
File encryption / decryption key
3
Resource module
Different types of content
2
Intermediary module
Index server

Figure 8. Functional modules of P2P architecture

The *client module* contains distributed computers of users, which are called peers.

The *resource module* contains different types of content, such as data, multimedia and other types of information. Each client computer stores the content that it shares with the rest of the P2P network.

The *intermediary module* represents the index server. The architecture of a P2P network can be decentralized, i.e. without a central index server or hybrid, i.e. with a central index server that maintains an index of the metadata for all files in the network. More specifically, a central index server maintains:

• A table of registered user connection information (IP address, connection bandwidth, etc.)

- A table listing the files that each user holds and shares in the network, along with metadata descriptions of the files (e.g. filename, time of creation, etc.)

In pure P2P network, peers contact each other directly (see Figure 7, arrow 1). In hybrid P2P network, a computer that wishes to join the network contacts the central server and reports the files it maintains (arrow 2).

The *access control module* contains the mechanism that uses file encryption and decryption keys. Having obtained the necessary information about the location of the required file, a peer requests the access to the file and using the file decryption key accesses the file (arrow 3).

Access control in P2P file-sharing networks

Typical P2P file-sharing systems do not emphasize on the access control. The primary objective in P2P networking was enabling free sharing between peers. Therefore they apply a simple access control mechanism, which is illustrated in Figure 8 as a file encryption and decryption mechanism. The authorized readers have the decryption key and the authorized writers have the signing (encryption) key. In Plutus [20] the reader receives a file-signature verification key, while the writers have a file-signing key. When the user wants to access the file to read or to modify it, he must have a key from the file owner. When the user wants to write, he must obtain the write token from the file owner. Using this token the writer can authenticate himself to the file server. The major drawback of this approach is the lack of efficient user revocation system. This brings the problem of re-encryption of large amount of data with a new key, when the reader leaves.

In Freenet [21] the files are encrypted with a random encryption key and the key is stored together with the file's identifier. This implies that any reader can access the file.

Similarities and differences with Fednets

PNs in a Fednet cooperate and share resources in a peer-to-peer manner and therefore, a Fednet is a peer-to-peer network of PNs. Consequently, there are similarities between Fednets and P2P file-sharing networks. First, the types of resources that are shared in Fednets and P2P networks are personal. Second, Fednets and P2P networks have high dynamism, i.e. the participants join and leave the network dynamically. Third, Fednets and P2P networks are both formed for a temporal sharing of resources.

However, there are also some differences between Fednets and P2P file-sharing networks. First, the administrative domains in Fednets are personal networks. Second, the scope of Fednets is broader than a file-sharing. Fednets can be created for a variety of applications for different purposes: emergency networks, learning environments, entertainment and business applications.

3.4 Wireless Community Networks

One of the possible applications of a Fednet is sharing the Internet access. Here we briefly compare Fednets with several other technologies for sharing the Internet access, such as Wireless community networks (WCN), P2P wireless networks confederation (P2PWNC) [22] and FON [23].

WCN is a development of interlinked community networks using wireless technologies. The goal of the WCN is to provide Internet access in areas where the conventional connection services are expensive or not available. WCN was developed by the Center for Neighborhood Technologies [4] to deliver low-cost, high-speed broadband access to homes, small businesses and community-based institutions. To join the WCN, a wireless networking equipment in a water-proof enclosure is installed on rooftops of the community, homes, apartments and other community buildings. This equipment is a wireless router running the mesh routing software. When the computer is connected to the router, it allows accessing the wireless community network. WCN is a mesh network, the wireless access points are interlinked to each other providing multiple and redundant paths, which makes the network robust to failures and damages.

The functional architecture

Figure 9 depicts the functional modules of WCN.

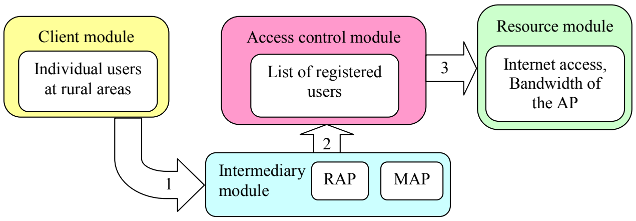


Figure 9. Functional modules of WCN architecture

WCN uses a mesh network to provide high-speed internet access to members of local communities. WCN consists of distributed WAP that belong to different owners. The *client module* contains individual computers that belong to different users.

The *resource module* contains the bandwidth of the Access Point to provide the Internet access.

The *intermediary module* contains Wireless access points: Main access points (MAP), with the direct connection to the Internet and Repeater access points (RAP), which pass the signal from a user until it reaches the Main access point. All access points are connected with each other in a mesh network and have the ability to wirelessly associate with each other without a landline connection between them. To connect to the Internet at least one Main access point is needed. The rest of the access points need to be within the signal range of another access point.

The *access control module* contains the filtering mechanisms based on MAC addresses at the access points or on higher levels based on the list of registered users.

Access control in Wireless Community Networks

WCN uses a simple access control mechanism, which is done at the wireless routers by configuring and MAC address filtering. A client attempting access must have its MAC address listed on an internal table of the wireless router (Figure 8, arrows 1 and 2). If so, it can be permitted to associate with the access point (arrow 3). In case the access point is a repeater, the traffic of the client will be forwarded to the next access point, in case the access point is the main access point (i.e. directly connected to the Internet), the client will get connected to the Internet.

Similarities and differences with Fednets

There are the following differences between WCN and Fednets. First, in WCN the administrative domains can be heterogeneous, e.g. individuals, institutions and organizations, while in Fednets the administrative domains are personal networks. Second, WCN have a relatively static backbone network, with static access points to the Internet. Participants join the network forming a mesh network on top of it. While a Fednet is a dynamic network, its composition, topology and the point of attachment of its components to the Internet can change over the time. Third, WCN are tailored for a specific application, i.e. sharing the Internet access, while sharing the Internet access is one of the possible applications of Fednets. Therefore WCN can be seen as a special case of Fednets.

3.5 P2P Wireless Networks Confederation

P2PWNC is a community of administrative domains that offer wireless internet access to each others registered users. It is a system that is built on WCNs and enables roaming of the users between WCNs based on incentive techniques. Reference [22] proposes a P2PWNC protocol. The goal is to simulate the participation in the WCN and the provision of 'free' Internet access to mobile users in order to enjoy the same benefit when mobile.

The functional architecture

Figure 10 illustrates the functional modules of P2PWNC. The *client module* contains institutions, service providers and operators.

There is no *TTP module* in P2PWNC. The system uses a reciprocity scheme, which does not require registration with authorities, and relies only on uncertified free identities and public/private key pairs.

The *resource module* contains the Internet access, bandwidth of the AP to access the Internet.

The *intermediary module* consists of the key entities in the P2PWNC, i.e. Domain Agents. Each independent domain maintains one Domain Agent, which has a unique logical name within the P2PWNC system. Domain Agents form a P2P network with each other. The Domain Agent has several functions, such as: name-service, authentication, accounting, consumer and provider strategy, service provisioning.

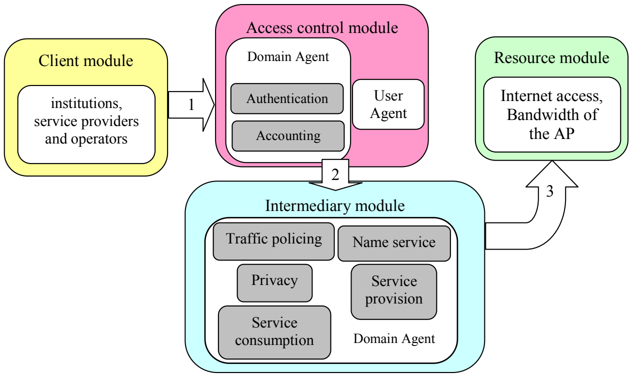


Figure 10. Functional modules of P2PWNC architecture

The main purpose of the Domain Agents is to eliminate the administrative overhead of roaming agreements. Instead of the roaming agreements, the Domain Agents use a token-exchange accounting mechanism. According to this mechanism a consumer Domain Agent transfers tokens to the visited Domain Agent in compensation for the used resources. So the central design goal is to build into the system incentive mechanism based on the reciprocal behavior: consumption and provision should be balanced.

The *access control module* contains Domain Agents and User Agents, which are explained in the next sub-section.

Access control in P2PWNC

The P2PWNC users can be registered with several domains, but they should have a unique identifier, in the form of '*user_at_domain*' for each account. For identity privacy, the users are allowed to have pseudonyms for each account.

The system uses a reciprocity scheme. Users sign digital receipts when they consume service. The receipts form a graph, which is used as input to a reciprocity algorithm that identifies the contributing users. Although the users can easily get free identities, the new users must first contribute to the system before using the services. The users are divided into teams. The contribution and consumption is evaluated on a team base. Therefore the scheme has a free-riders problem.

In order to use the services the users input their user identifiers and associated security credentials to user agents (Figure 10, arrow1). The user agents carry out the authentication procedure in cooperation with the Domain Agent. The users may use different identifiers, choosing from the Domain Agent who has a higher token level. When the access is granted (arrow 2), the Domain Agent coordinates the wireless service provisioning and consumption for its domain.

In every domain (e.g. institution, service provider) there is an associated group of registered users. The Domain Agents maintain the list of its own registered users. The Domain Agent is an economic agent within the P2PWNC, it is responsible for the coordination of bandwidth consumption by the registered users of the domain in a roaming scenario and for the coordination of bandwidth provisioning by the domain itself.

Similarities and differences with Fednets

There are the following differences between P2PWNC and Fednets. First, in P2PWNC the administrative domains can be heterogeneous, e.g. individuals, institutions, service providers and operators. Second, since it is built on top of WCN, P2PWNC have a relatively static backbone network, with static access points to the Internet. Third, P2PWNC are tailored for a specific application, i.e. sharing the Internet access. Therefore similar to WCN, P2PWNC can be seen as a special case of Fednets.

3.6 FON

FON [23] is a system of shared wireless networks. The FON's members share their WiFi with others, in return they can freely access all other FON wireless access points that are available all over the world. This is achieved by sharing the bandwidth of their special routers, called La Fonera routers.

The functional architecture

Figure 11 illustrates the functional modules of FON.

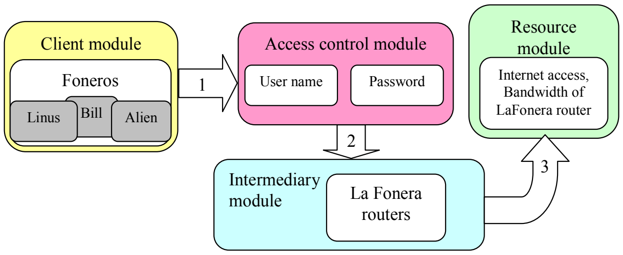


Figure 11. Functional modules of FON architecture

The *client module* contains the users of the FON network, called Foneros. Foneros are distinguished into three types based on their membership: Linuses, Bills and Aliens, as shown in Figure 10. Linuses and Bills are registered users of the FON network. They share their home WiFi hotspot with the FON network and can use any FON hotspot for free, can roam the FON network for free. Aliens do not share their bandwidth but they can use the FON network by purchasing daily passes. FON passes are similar to prepaid cards. Aliens can purchase FON passes by detecting a FON signal and connecting to FON or by sending SMS through their mobile phones. Aliens can also get 15 minutes of free WiFi access to any FON spot per day.

The *resource module* contains the bandwidth of La Fonera routers, which is shared between Foneros. FON consists of distributed La Fonera routers that belong to different owners. While roaming, the user can connect to internet by means of these routers through the WiFi available in the vicinity. Using laptops or WiFi enabled devices, such as phones, cameras, Foneros can access any FON spot around the world.

The *intermediary module* consists of La Fonera routers of Foneros.

The *access control module* in FON contains the authentication mechanism based on FON usernames and passwords.

Access control in FON

FON software includes a level of access control, which could be beneficial for WiFi in open network with little or no security, also beneficial for service providers. All registered members of the FON network have FON username and password. Once an Alien has registered with FON (Figure 11, arrow 1), using its user name and password, it can be granted the access to the Internet through La Fonera router (arrow 2), which provides with the part of its bandwidth for the traffic of the Alien (arrow 3).

Aliens can also use their FON username and password to access their own personal User Zone. In the User Zone, the Alien can retrace her WiFi activities through the FON Community. Including seeing how many FON Passes they have purchased, used and how many they still have remaining.

Similarities and differences with Fednets

There are the following differences between FON and Fednets. First, in FON the administrative domains are individuals with *La Fonera* routers. Second, FON have a relatively static backbone network, with static access points to the Internet. Participants join the network forming a mesh network on top of it. Third, FON are tailored for a specific application, i.e. sharing the Internet access. Since sharing the Internet access is one of the possible applications of Fednets, FON can be seen as a special case of Fednets.

3.7 Comparison of related technologies

In this section we summarize our survey on resource sharing technologies and paradigms in Table 1.

Table 1. Comparison of Fednets and related technologies

Technology	Definition	Typical applications	Administrative domains	Scale of the system	Shared resources	Access control mechanisms
Grids	Hardware and software infrastructure to allow coordinated resource sharing and problem solving	• Medical/Healthcare • Bioinformatics • Nanotechnology • Engineering • Natural Resources and the Environment	Individual users, organizations, Virtual Organizations	Number of services, participants and geographic scale is **large**	Hardware, software, computer processing power, big scale computing and data storage facilities	**Centralized** access control (grid mapfile, CAS, VOMS, PERMIS, PRIMA), **distributed** access control at stakeholders (AKENTI)
Secure Virtual Enclaves	An infrastructure implemented in middleware to allow multiple organizations to share their distributed data and application objects	• Military environments • Disaster or incident response teams • Business environments	Organizations	Number of services, participants and geographic scale is **small**	Distributed application objects, information and data of organizations	**Distributed** access control to the SVE, **distributed**, local access control to the SVE resources
P2P file-sharing networks	A collection of networking nodes where each node has equal responsibilities and capabilities in providing and consuming the services.	File and content sharing (e.g. Napster, Gnutella, Fasttrack, Morpheus and Kazaa)	Individuals users, organizations	Number of participants and geographic scale is **large**. Number of services is **small**.	Files, information, media and entertainment	**Distributed** access control with encryption/decryption keys
WCN	Interlinked community network using wireless technologies	Low-cost broadband connectivity and related opportunities such as job searching capability and skill development, to underserved households, community groups, and small businesses.	Individuals, organizations	Number of participants is **large.** Number of services and geographic scale is **small**.	Wireless Internet access is shared, by means of sharing access point repeaters for traffic forwarding between neighbors	**Distributed** access control at Wireless access repeaters and Wireless access points by MAC address filtering, or on higher levels with the list of registered users
P2PWNC	System that is built on WCNs and enables roaming of the users between different WCNs based on incentive techniques	Universities, residential hotspots, private companies that provide WLAN access to employees, mobile operators offer wireless internet access to each others registered users.	Individuals, organizations, service providers, operators	Number of participants and geographic scale is **large.** Number of services is **small**.	Wireless Internet access is shared on the basis of reciprocity algorithm.	**Centralized** access control to the domain carried out by the Domain agents

Table 1 (continued). Comparison of Fednets and related technologies

Technology	Definition	Typical applications	Administrative domains	Scale of the system	Shared resources	Access control mechanisms
FON	A system of shared wireless networks	Sharing personal WiFi with others, in return to the possibility to freely access all other FON wireless access points.	Individuals, called 'Foneros' with their home WiFi and La Fonera routers	Number of participants and geographic scale is **large.** Number of services is **small.**	Part of the bandwidth is shared to give the Internet access to others	**Centralized** access control with password and user name submitted to the FON special site
Fednets	Ad-hoc, temporal, secure cooperation of independent Personal networks	• Family networks for entertainment and remote file sharing • Ad hoc network during the Project-meetings • Inter-vehicle networks to share information on the road conditions • Emergency, disaster relief and rescue/recovery networks to rescue people • Health-care and hospital networks • Distance learning networks and virtual classrooms • Commercial resource sharing • Online gaming • Networks for information services	Individuals with their personal networks	Number of services, participants and geographic scale is **large**	Personal resources and services (audio-video, storage, printing, processing, routing, internet access etc.)	**Centralized** access control to the Fednet, **distributed**, local access control to the Fednet services at PNs

We observe from the table, that all systems, except SVE, may have administrative domains composed of individuals participating with various types of resources. This observation suggests that the group networking formed between individuals is of particular interest.

As can be seen from Table 1, all systems differ in their scalability. Among all others, Fednets and grids can have a large number of service types, participants and their geographic scale is large. In P2P file-sharing networks, P2PWNC, and FON despite their large number of participants and large geographic span, the number of shared service types is small. In WCN, the geographic scale is also small as well as the number of shared services. Furthermore, in SVE, in addition to the geographical scale and number of shared services, the number of participants is also limited, since the SVE is a closed, controlled collaborative network.

Furthermore, our survey reveals that each system is designed for a particular application for sharing specific types of resources. Access control is the most essential component in service provisioning in a cooperative environment. The type of application and shared resources are important when choosing for specific access control architecture. The majority of systems deploy distributed access control to shared resources and services.

From the surveyed technologies grids are of particular interest, because based on the structure, Fednets can be seen as '*a grid of personal networks*'. In addition, grids and Fednets have similarities with regard to their scalability in geographical span, number of service types and number of participants. Moreover, grids and Fednets combine both centralized and distributed approach in their access control architectures. This determined our motivation to survey a number of access control mechanisms used in grids environments. Section 4 is devoted to this topic.

4. ACCESS CONTROL TO SHARED RESOURCES

Further in our survey we focus on different approaches for access control in grid environments. We analyze them based on the IETF Authentication Authorization and Accounting framework [24] (AAA), which is the authorization framework for Internet resources and services.

4.1 Generic AAA Framework

Basic conceptual entities

The basic conceptual entities that may take part in the authorization process are illustrated in Figure 12. They are:

- Users,
- User home organizations, with its AAA Server,
- Service providers, with its AAA Server and Service Equipment.

AAA server is a network server used for access control. The user home organization based on the user agreement checks whether the user's request for a service should be permitted. This task is performed by the *AAA server of the user home organization*.

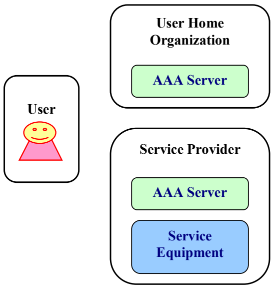


Figure 12. The Basic Authorization Entities

When the user's service request gets to the service provider, the *AAA server of the service provider* authorizes the user access to its service based on the agreement with the user home organization. The service equipment of the service provider is the one that provides the service to the user.

The framework defines several authorization message sequences to achieve trust between the user and the service provider. There are two cases: a single domain case and a roaming case. In a *single domain case* the user, the service provider's AAA server and the service provider's service equipment take part. No user home organization is involved. The *roaming case* explores the situation where the organization that authenticates and authorizes the user is different from the organization providing the services. This means that in this case, both user home organization and service provider are involved with their AAA servers.

In group-oriented networks individual users communicate with each other without involving the user home organization. Therefore, further we describe the sequences of authentication message flow for a single domain case. There are three message exchange sequences defined in AAA framework between the user and the service provider. They are push, pull and agent sequences. We briefly describe them here.

The Push sequence

Figure 13 depicts the push sequence. The user gets a ticket or a certificate from the service provider's AAA server (arrows 1 and 2) and then presents it to the service provider's service equipment together with the request (3). The service equipment uses the ticket to verify that the request is approved by the service providers AAA server. By the successful verification, it grants the access (4).

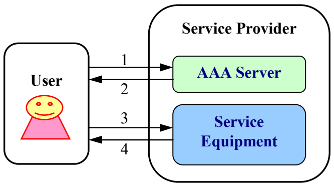


Figure 13. Push sequence of authorization message flow

The Pull sequence

This sequence is typically used in dial-in applications. The user sends the request to the service equipment (arrow 1), which forwards it to the service provider's AAA server (2), this is illustrated in Figure 14. The AAA server evaluates the request and returns the response to the service equipment (3). The service equipment sets up the service and notifies the user that it is ready to serve (4).

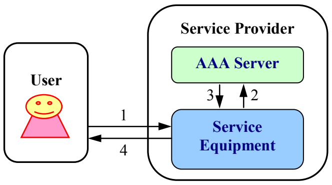


Figure 14. Pull sequence of authorization message flow

The Agent sequence

In this sequence the service provider's AAA server acts as an agent between the user and the service equipment, as is depicted in Figure 15. It receives the request from the user and sends authorization and configuration information to the service equipment.

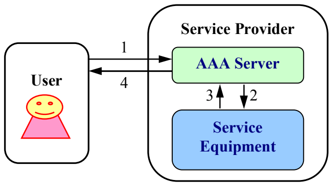


Figure 15. Agent sequence

AAA framework and policy framework

IETF RFC 2904 [24] also describes the relationship of authorization and policy. It extends the policy framework presented in IETF RFC 2753 [25] to support policy across multiple domains. RFC 2904 introduces the components such as the *Policy Decision Point (PDP)*, which makes access control decisions based on policies; and the *Policy Enforcement Point (PEP)*, which enforces the decisions made by the PDP.

When mapped into the policy framework, the AAA server locates the PDP function. The PEP function is located at the Service Equipment of the Service Provider, as is shown in Figure 16.

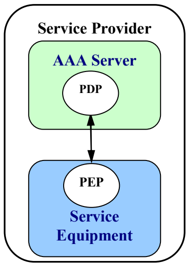


Figure 16. Mapping of policy framework components to AAA framework

With policy extension, the above-mentioned sequences will be as follows:

1. *Push sequence*. The user calls the PDP, the PDP returns the authorization decision and the user submits it to the PEP.

2. *Pull sequence.* The user calls the PEP. The PEP pulls the authorization decision from the PDP and based on this decision it grants or denies the access.

3. *Agent sequence*. The user calls the PDP. The PDP sends the user request along with the authorization decision to the PEP. This way the PDP acts as an agent on behalf of the user.

4.2 Access control architectures for grid environments

A number of different authorization architectures are reported in literature for access control in distributed environments, such as grids. They are based on different approaches, such as Certificates (CAS [14], VOMS [15], AKENTI [17], and PERMIS [16]), Signed assertions (CAS), Capabilities (CAS), Roles (PERMIS and PRIMA [18]) and Policy statements. In this section, we describe these authorization architectures and the control over the access to shared resources based on the message exchange patterns described in Section 4.1.

4.2.1 CAS

To address the scalability of the access to the distributed virtual community's resources and improve the manageability of user authorization, a trusted third party - a community authorization service (CAS) was proposed [14] in 2002. It minimizes the burden of maintaining grid map files (discussed in Section 3.1) by the administrators.

Reference [26] discusses a number of challenges imposed by Virtual Organizations (VO), such as scalability (the cost of administering a VO, adding and removing participants, changing community policy increases by growing of the VO) and complex policy hierarchies. The CAS architecture is built on Public Key Infrastructure (PKI) [37] and Globus Toolkit Grid Security Infrastructure (GSI) [34] and addresses the issues of single sign on, delegation and scalability that arise in Virtual organizations (VO). According to the CAS principles, the community delegates the access granting rights to a subset of its resources to the central authority, i.e. the CAS server. When the client requests to use the resource of the community, the CAS server, based on the policies defined by the communities, decides about the access rights that should be granted to the user. Having taken the decision, the CAS server produces self-signed certificates with permissions to access the resources. The access to the resource will be granted, if the validity of the certificate and the CAS service is proved.

Access control

CAS is an authorization service developed within the Globus project for Grid environments. CAS server acts as a trusted intermediary between the VO users and resources. The resource owners grant the access to the subset of their resources to the VO. The CAS in this VO is a trusted intermediary between the users and resources, which decides who can use the resources. This means that the resource owner delegates the allocation of authorization rights to the CAS server.

First, the user becomes a member of the community. This process corresponds to the first-level access control. Afterwards, the user can request a service by contacting the CAS server, which delegates the rights to use the requested service that belongs to the community. The rights are in the form of capabilities. They are embedded in GSI proxy credentials as policy assertions written in SAML [36] and signed by the CAS server. Having obtained the proxy credentials, the user presents them to the resource to access the resource on behalf of the community. This process corresponds to the second-level access control.

Authorization message flow

Figure 20 illustrates the resource access using CAS. The CAS server stores the policies which contain the list of objects and their rights. This information is included in the extension of the delegated proxy certificate.

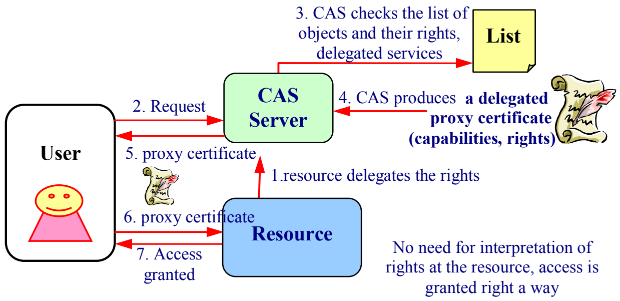

3. CAS checks the list of objects and their rights, delegated services
List
2. Request
CAS Server
4. CAS produces a delegated proxy certificate (capabilities, rights)
User
5. proxy certificate
1.resource delegates the rights
6. proxy certificate
Resource
7. Access granted
No need for interpretation of rights at the resource, access is granted right a way

Figure 20. Resource access using CAS. Push sequence

The requestor contacts the CAS server to get a delegated proxy certificate that includes the information about what resources can be accessed and to what extent. The delegated proxy certificate is a short-lived X.509 certificate. To access the resource the user submits this proxy certificate to the gatekeeper of the resource. This certificate is enough to access the resource, there is no need to submit the attribute certificate (attribute value bindings to a user) to the gatekeeper. This means that the granting the access rights to the community resources is done in advance, before the user contacts the gatekeeper. This offers some relief to the resources from interpreting the rights of the users. This approach corresponds to a push sequence of the AAA framework.

The CAS administrator is responsible for adding each user to the appropriate group of the community. The CAS administrator can delegate to others the administration of subsets of objects. Here note, that the member administration is centralized, with the delegation possibilities.

Observations

The CAS approach has the following drawbacks. CAS issues a proxy certificate instead of the attribute certificate and the authorization information is included in the extension of the proxy certificate. The extension includes the restriction on the access rights, placing specific limits on the rights of the user. When the service receives the certificate it should check the extension to know the restrictions to the access rights. This approach is not efficient, because it requires modification at the service side. Furthermore, CAS does not support roles, but permissions to do actions. Consequently, the CAS server records permissions and does not record the roles, because the roles of the users or the groups of the users are not defined.

The drawbacks of the CAS approach also include the requirement for enforcement within the application code, Policy Enforcement function is built into the grid service application, so there is a need for a trusted application code. Moreover, a group owned infrastructure component – CAS server and a community administrator are required. This also raises scalability problems.

4.2.2 VOMS

VOMS (Virtual Organization Membership Service) [15] is another implementation of the access control to the grid resources. It provides the authorization information about members and an authentication and authorization service within the VO. It is developed in the European Data Grid project [27].

VOMS delegates the authorization of the users to the managers of the VO and allows managing user roles and capabilities centrally. The main difference between the approaches in CAS and VOMS, is that in VOMS the resources should carry out the interpretation of rights based on the membership certificates of the users, while in CAS the resources do not need such interpretation, since the certificate is enough to access the resource.

Access control

VOMS provides the authentication and authorization services, we can map its architecture to the AAA framework. The architecture of the VOMS uses the authentication and delegation mechanisms provided by Grid Security Infrastructure (GSI). In VOMS the authorization to the resources is based on policies, which are written by the VOs representing their agreements with the resource providers. VOMS embeds attribute certificates in GSI proxy credentials that specify group and VO membership information for access to community resources. The requestor contacts the VOMS server to get a credential and submits this credential to access the resource.

The authorization information is separated into two types, because this information controls the resource access in VO from different perspectives, with different roles.

First, the information regarding the relationship of the resource user to the VO, for example its membership, belonging to which group and etc, this information is stored at the server managed by the VO;

In addition, the information regarding the relationship of the resource user to the resource provider, for example, what the user can do at the resource and etc., this information is stored locally at the resources.

Here we encounter two different types of access control, which reminds the two-level-access control that we defined for Fednets. The first is the access control to the community as a member. The second is the access control to the resources of the community.

Authorization message flow

In the first version, VOMS was a system for dynamically creating grid map files from LDAP directories containing the details about the VO users. A grid map file contains the list of authenticated distinguished names of the grid users mapped into the corresponding local user accounts names. The resources could periodically retrieve them to make authorization decisions, as is shown in Figure 18. This approach corresponds to a pull sequence of the AAA framework.

This approach maximizes the work of the resource administrator, because he must first pre-configure the grid application with the names of every VO user, if the user is allowed to access the grid resource. This approach is not scalable and not flexible, and the administrative task can not be distributed throughout the VO.

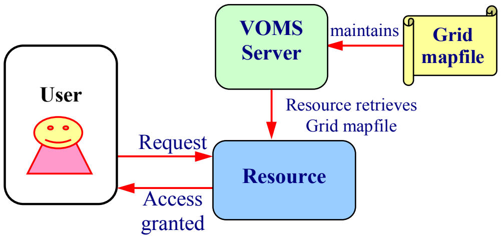


Figure 18. Resource access using VOMS. Pull sequence

Later version of the VOMS issues a short-lived X.509 Attribute Certificate [28] for the VO users which they can submit to the resource. The certificates are signed by the VOMS server. The certificate contains the information about the users, such as local account name, to which group does the user belong to, what roles the user is assigned within this group and some other privileges and capabilities. Therefore the resources do not need to retrieve a grid map file, since all necessary information to verify the identity is included in the certificate. However, the resource needs software to interpret the attribute certificate. This approach corresponds to a push sequence of the AAA framework and is illustrated in Figure 19.

Observations

VOMS has a community centric attribute server that issues authorization attributes to members of the community, similar to CAS server. But in CAS the subjects have a group credential, while in VOMS subjects authenticate with their own credentials.

The drawback of the VOMS approach is that the resources should carry out the interpretation of rights based on the membership certificates of the users. This puts a burden to the resources, since the resources should know how to do the interpretation. Furthermore, VO administrator maintains a centralized database to add each VO user and gives users appropriate attributes needed to access the VO resources. This approach has scalability problems, in managing joining and leaving members of the VO, their access rights and roles within the VO, since it is based on centralized model in user management.

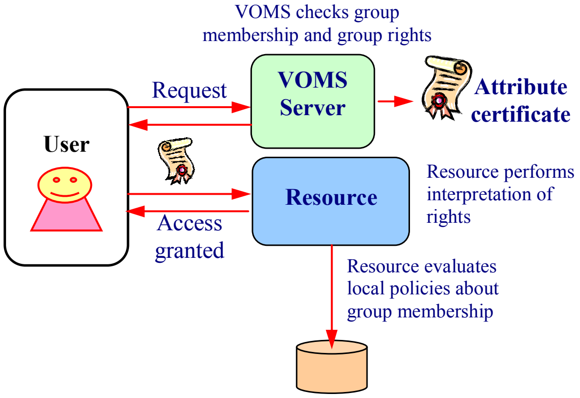

VOMS checks group membership and group rights
Request
VOMS Server
Attribute certificate
User
Resource
Resource performs interpretation of rights
Access granted
Resource evaluates local policies about group membership

Figure 19. Resource access using VOMS. Push sequence

4.2.3 AKENTI

AKENTI [17] is a distributed policy-based authorization system for grid environments and is designed for authorizing the access on web resources, such as web sites. AKENTI addresses the issues of providing restricted access to resources that are controlled by multiple stakeholders. The Stakeholders in AKENTI are the parties with authority to grant access to the resource.

Access control

AKENTI does not require any central authority to enforce the access control to the resources. AKENTI uses distributed policy certificates in XML format. These certificates are signed by the stakeholders from different domains, who decide on the access control to a resource and place its own restrictions to the usage of the resource. AKENTI makes a dynamic authorization decisions based on supplied credentials and applicable usage policy statements defined in AKENTI policy language. For expressing policies and certificates AKENTI uses XML, although the first version of AKENTI used a simple keyword language.

Digitally signed certificates specified in AKENTI can contain the following information:

- identity authentication information,
- attribute certificates,
- use condition certificates (the list of users owning the attributes and the explanation of which attributes are needed for which access rights),
- policy certificates (include the list of trusted CA and stakeholders and the links from where the use-conditions and attribute certificates can be retrieved).

Authorization message flow

AKENTI can use both pull and push models of authorization information flow. Figure 22 illustrates the push model. In AKENTI there can be multiple stakeholders participating and administering one resource. A new user should contact the stakeholder to be added to the list of resources and appropriate policy files. This process corresponds to the first-level access control.

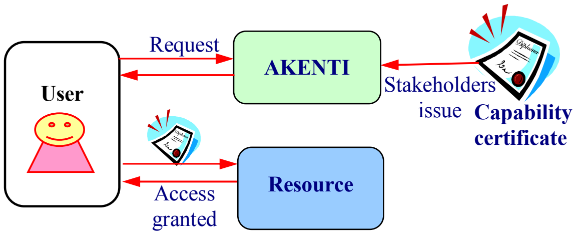


Figure 22. Resource access in AKENTI. Push sequence

To access the resources the requestor is required to present some credentials to AKENTI authorization system. First, AKENTI authenticates the user based on the X.509 identity certificates, then it uses the attribute certificates belonging to the user and use-condition certificates (regarding the resources) and makes a decision on the access rights of the user. The system then provides the client with a capability certificate. The client contacts the gatekeeper of the resources by presenting this capability certificate.

Observations

In AKENTI, the resources are accessed based on the resource access policies. The evaluation of policies and granting the access rights are done after the gatekeeper of the resource is contacted, whereas in CAS, the rights are already included in the capabilities issued by the CAS server.

AKENTI does not require a centralized authority to run the access control policies. Although the resources are owned and controlled by multiple stakeholders, the access to the resources is controlled by means of distributed policy certificates, without a central authority.

The drawback of AKENTI is that it gives the user the total access, not a fine-grained access to the resources. As a consequence, the user's access can not be limited during the sessions. Moreover, AKENTI does not link the identities with groups or roles but with permissions, therefore the user can not specify the role that he wants to use during the access. As a result, the attributes in AKENTI cannot form a role hierarchy.

AKENTI specifies separately the authorities for performing authentication and for creating and signing attribute certificates. This introduces another drawback that the resources must know about the CA of each user, which causes scalability problems. In contrast, in CAS the resources must know only the CA of the CAS server.

4.2.4 PRIMA

PRIMA [18], [29] is a system for Privilege Management, Authorization and Enforcement in grid environments to support dynamic, spontaneous, short-term collaborations of small groups of grid users. While CAS and VOMS are systems that rely on central servers for the authorization service in grids, PRIMA is a fully decentralized system that enables direct trust establishment among participants. It supports dynamic authorization policies for grid resources. PRIMA distinguishes from other authorization systems by its support for the creation, configuration and management of user accounts on demand. Other grid security services support only static accounts, which limit the scalability, hinder collaboration and creates security holes through static accounts.

Access control

PRIMA focuses on access control for small and dynamic working groups. The system uses fine-grained privileges as fine-grained access rights. It uses privileges to enforce policy statements. The subject privileges are issued by resource owners and administrators, or group and project leaders. Both privilege statements and policy statements are expressed in XACML and are embedded in X.509 Attribute Certificates [28], [30], so the X.509 Attribute Certificate carries privilege and policy statements.

In PRIMA the privilege attributes are issued by individual attribute authorities, such as project leaders, resource owners, administrators, but not community servers like in VOMS or CAS.

Regarding the levels of the access control, the first-level is carried out when the entity becomes a group member. The PRIMA is used for the second-level access control.

Authorization message flow

PRIMA implements a hybrid model of authorization message flow as is shown in Figure 21. Although the user pushes the acquired privileges to the resource (i.e. a push sequence), still the resource requests the access control decision from a PDP function based on these privileges. Therefore this part of the message exchange corresponds to the pull sequence.

The privileges are collected by the Policy Enforcement Point (PEP) and are checked against the access control policies at the Policy Decision Point (PDP). PDP returns an authorization decision to the PEP and a set of recommendations on the actions, for example, setting up a local account based on the valid privileges, file access permissions, network access and etc.

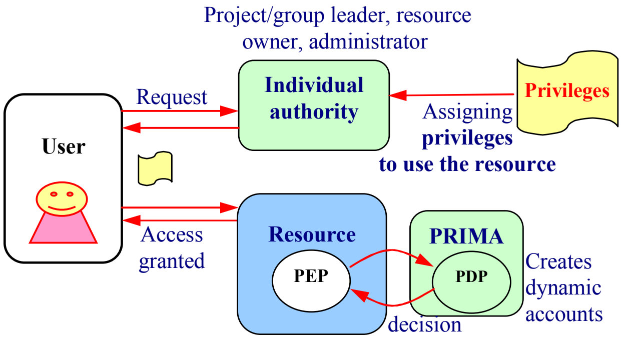


Figure 21. Resource access in PRIMA. Hybrid model

Observations

PRIMA is distinguished from other authorization systems with its support for the creation, configuration and management of user accounts on demand [18], [29]. Dynamic accounts are like dynamic IP addresses. They are taken from the pool of available addresses and returned into the pool when released. The pool of dynamic accounts is created by the system administrator. These accounts do not allow direct login and have minimal rights. When a dynamic account privilege is presented, the system first checks for an existing account matching the distinguished name and optional project identifier, then maps the user to the existing dynamic account. If the user is not found in the map, a new dynamic account is assigned from the pool of available accounts. Before expiring the holder is notified by the privilege revocator. Once the account is expired, it is reset and returned to the pool.

In PRIMA individual privileges can be grouped to form a group, the group of privileges can be applied to a set of users – holders of the roles. PRIMA is different from the Role-Based Access control (RBAC) system, because RBAC system creates the roles and binds the access rights to roles. Besides RBAC focuses on the administrators and resources, while PRIMA focuses on the resource users.

For the implementation PRIMA module is integrated with the Globus toolkit as an authorization component. PRIMA module acts as a PDP and makes fine-grained authorization decisions based on the privileges of the user.

Creating dynamic user accounts on demand allows the users to utilize the resources on temporal basis. The accounts are created based on the privileges of the users assigned by the authorities, such as resource owners, project leaders or administrators. Dynamic accounts can be replaced by static accounts on demand. This gives flexibility in managing the user accounts.

The drawback is that the resources need extra functionality to implement the PDP and dynamic account creation.

4.2.5 PERMIS

PERMIS [16], [31] is an authorization system that implements a Role Based Access Control mechanism for different role-oriented scenarios. A user is granted rights to access a resource based on the authorization policy for the resource, and a set of role attributes that the user possesses.

A user's attributes are stored in digitally signed X.509 Attribute Certificates [28]. Given the name of the user, PERMIS retrieves the user's attributes/roles and makes decisions based on them. The authorization policy, written in XML, expresses which users can be assigned what roles by whom, and what privileges are bound to each of the roles. The XML policy is then inserted in an X.509 Attribute Certificate, signed by the manager who wrote it, and stored in an entry in an LDAP server.

Access control

When an application starts up, its PEP passes to the PERMIS PDP the name of the manager, the location of the LDAP directory, and the unique number of the policy to be used. Each policy is assigned a globally unique number, so that a manager can create different policies to be used in different contexts. Then the PERMIS PDP retrieves the policy X.509 Attribute Certificate from the LDAP directory, checks the signature and the policy number. If both are correct, PERMIS makes the authorization decision evaluating this policy based on the attribute/roles of the user retrieved from the X.509 Attribute Certificate. Therefore, PERMIS is considered as a strong policy engine to control the resource access.

PERMIS consists of two subsystems: privilege allocation and privilege verification. The first issues the attribute certificates and stores them in LDAP, the second retrieves the attribute certificates and the policies on the user roles from a pre-configured list of LDAP directories. In PERMIS the entity that creates policies is called a Source of Authority.

Similar to PRIMA, PERMIS does not include the first-level access control. PERMIS grants the access to the resources based on the roles of the users and the resource policies. This process corresponds to the second-level access control.

Authorization message flow

PERMIS does not provide an authentication services, but it can work with any authentication system, such as Shibboleth [32], Kerberos [33], PKI [37] or username/password. Given a username, a target and actions, the PERMIS, based on the policy, decides whether the user is granted or denied the access to the resource.

Figure 23 depicts the resource access in PERMIS, which corresponds to the pull sequence of the authorization message flow. However, the users can also push the certificates to the system for verification. Therefore, PERMIS uses a hybrid model of authorization.

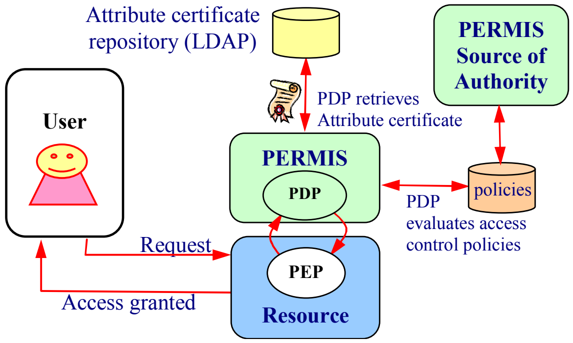

Attribute certificate repository (LDAP)
PERMIS Source of Authority
User
PDP retrieves Attribute certificate
PERMIS
PDP
policies
PDP evaluates access control policies
Request
PEP
Access granted
Resource

Figure 23. Resource access in PERMIS. Pull sequence

Observations

PERMIS is an alternative to VOMS. Users are given roles and attributes that belong to these roles. The roles and attributes are assigned permissions to access the resources. It is also called a privilege management infrastructure that uses X.509 certificates.

Similar to CAS and VOMS, PERMIS also uses the attribute certificates, which are stored in the repository for attribute certificates. After the user is authenticated successfully, the system retrieves the attribute certificate of the user from the repository. To make decisions, PERMIS processes the content of the policy file and the content of the attribute certificate of the user.

4.2.6 My Proxy

Most Grid Portals (gateways) require that the user delegates the rights to the server to act on its behalf. Normally the Grid resources are protected by GSI [34], which supports such delegation. But web security protocols do not support the delegation function, so this leads to incompatibility between Grid security and Web security. To address this problem the reference [35] proposes an online credential repository system called ‘MyProxy’, which allows smooth operation of grid portals that use GSI to interact with grid resources. MyProxy is open source software for managing X.509 PKI security credentials, such as certificates and private keys. MyProxy combines an online credential repository with an online certificate authority to allow the users to securely obtain credentials when and where needed.

Authorization message flow

Figure 24 shows the process of accessing the grid resources through web portals using MyProxy credential repository.

Resource access using MyProxy corresponds to the pull sequence of the authorization message flow. By the request of the user to access a grid resource, web portal retrieves user credentials from MyProxy repository. Using these delegated credentials web portal authenticates to the grid resources and provides the user with the access to this resource.

The first-level access control is the process when the user becomes a grid user or a VO member. MyProxy provides credentials that are used for the second-level access control, i.e. to access the grid resources.

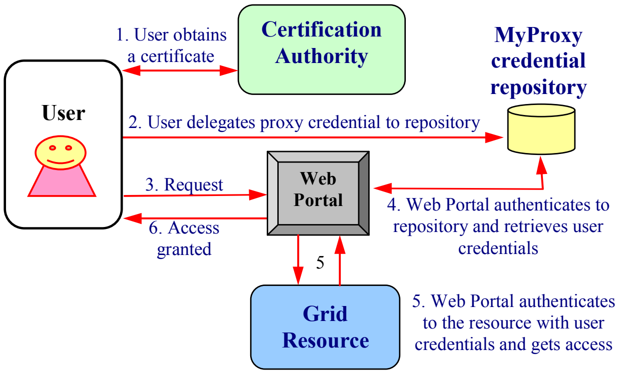

1. User obtains a certificate
Certification Authority
MyProxy credential repository
User
2. User delegates proxy credential to repository
3. Request
Web Portal
4. Web Portal authenticates to repository and retrieves user credentials
6. Access granted
5
Grid Resource
5. Web Portal authenticates to the resource with user credentials and gets access

Figure 24. Resource access using MyProxy. Pull sequence

Observations

MyProxy is a system to provide online short-lived credentials to access grid resources. MyProxy supports multiple authentication mechanisms, including passphrase, certificate, Kerberos, VOMS, LDAP and One Time Passwords. The MyProxy CA issues short-lived session credentials to authenticated users. The repository and CA functionality can be combined into one service or can be used separately.

4.3 Summary of access control architectures

In this section we summarize the discussed access control architectures. Table 2 provides an overview of each of the systems, in terms of the entities of their access control architecture, what information these entities use to carry out the access control and how this information is conveyed between these entities.

Table 2. Comparison of access control architectures discussed in this paper.

Access control architecture	**Description**	**entities of the access control arcthitecture**	**Access control information**	**Authorization sequence**
CAS	An **authorization** system for distributed virtual community resources	CAS server Organizational resources Users	Delegated proxy certificate, roles, policies, capabilities, group credential. Access rights are in the proxy certificates in a form of SAML policy assertions	**Push** The user gets the attribute certificate from the CAS server and then presents this certificate to the resource
VOMS	An **authentication** and **authorization** system for virtual organizations	VOMS server Organizational resources Users	Attribute certificates, policies, group membership, roles in the group	**Pull** When user requests a service, the resource retrieves a grid map file to make authorization decisions **Push** The VOMS server issues a short-lived X.509 Attribute Certificate for the VO users which they can submit to the resource.
PRIMA	Privilege management and **authorization** system for dynamic small groups of grid users	Project leaders, administrators Grid resources Small group of grid users	Privileges, policies, X.509 Attribute Certificates, dynamic accounts. Privileges are embedded in X.509 Attribute Certificates	**Hybrid** The user pushes the acquired privileges from the PRIMA system to the resource (i.e. a push sequence). The resource requests the access control decision from a PDP function based on these privileges (i.e. a pull sequence).
AKENTI	Distributed policy-based **authorization** system for web resources, grids	Stakeholders Resources of multiple stakeholders Users	Capability certificates, policies, use conditions for resources, Mutual authentication using X.509 certificates	**Pull** The user contacts the resource, which calls AKENTI with the user name and the resource name. Then the resources obtain an access control decisions. **Push** AKENTI gives a capability certificate to the user. To access the resource the user presents it to the gatekeeper of the resources.
PERMIS	An **authorization** system with role-based access control system	Source of authority (entity that creates policies) Resources Roles	Roles, privileges, policies, X.509 Attribute Certificates	**Pull** The PDP contacts the attribute certificate repository to retrieve the appropriate certificate and then runs the access control policies. Decision is a Boolean grant/deny response. **Push** Users can push the certificates to the PDP for verification and access rights.
MyProxy	An online **credential repository** to bridge the incompatibility between web- and grid-portals	MyProxy online credential repository Web portal Web resources Users	X.509 Attribute Certificates, delegated proxy credentials, MyProxy credential repository	**Pull** Web portal retrieves user credentials from MyProxy credential repository and grants the access based on them.

As can be seen from the table, most of the architectures use X.509 Attribute Certificates to store the authorization information. Several mechanisms use both pull and push models of authorization message flow (e.g. VOMS, AKENTI and PERMIS). PRIMA deploys hybrid model, since the authorization contains consecutive push and pull sequences. Moreover, all systems use access control policies based on different conditions (e.g. roles, attributes, privileges, capabilities). PERMIS among them is a strong policy engine to control the resource access based on roles and privileges. Access control decisions can be in a simple 'grant/deny' form (e.g. AKENTI), as well as fine-grained access differentiating from total till restricted access to the resource based on various criteria (e.g. PERMIS).

We can recognize some similarities between our approach and other approaches. For example, the Fednet manager issues a *membership credential* which in case of VOMS is a 'VOMS certificate' issued by the VOMS server. To request the service, the client PN presents this membership credential to the service providing PN. Based on the *access control policies defined by the PN owner*, the access control decision is made.

Furthermore, *common services* in a Fednet remind the CAS principles. As was explained in Section 2.3, the common services of a Fednet are accessible to all members of the Fednet upon presenting their 'membership credentials' issued by the Fednet manager. In the case of CAS it is a 'proxy certificate' issued by the CAS server, which grants the access rights to the community resources.

Finally, dynamic access control. For the second-level access control the PN owners define their own *policies* and *access privileges* to allow the access to their personal resources. The membership class is assigned by the Fednet manager based on the previous experiences, the contributions, the reputation or the role of the Fednet member. Different membership class corresponds to different privileges to access the service. Privileges together with the membership class dynamically change the access rights to the Fednet services. This approach reminds the dynamic authorization policies provided in the PRIMA system.

All approaches have their attractive points. For example, the approach taken in VOMS facilitates the management, since there are dedicated VOMS administrators for this task. The administrators provide the user with the authorization credentials that are interpreted by the resource. Furthermore, the approach taken in CAS is attractive with its proxy certificates, which give the access to the resources, so that the resources do not need the interpretation of the credentials. An interesting part of MyProxy approach is that online credential repository acts as a trusted intermediary between the web-portals and grid users. PRIMA approach is interesting with its dynamic on-demand creation and management of user accounts for small groups of grid users. PERMIS might be attractive with the creation of role-hierarchies and fine-grained access control based on roles and policies.

5. SUMMARY

In this paper, we described Fednets, which are group-oriented networks to share personal resources and services to achieve a common objective. We introduced its architecture in functional modules, which provide explicit information about what is shared, who is sharing and how the sharing is done. Further in our survey we compared Fednets with the related technologies in order to place Fednets amongst them. We summarize our observations as follows:

The access to the system, i.e. first-level access control. In SVE a new enclave can join the SVE through voting if only the majority of the enclaves agree on that. Each enclave maintains the list of trusted collaborators, i.e. enclaves of other organizations. The same principle works for VOs. In P2PWNC the first-level access control is carried out by Domain agents and in FON it is carried out by a special registration site. In Fednets, the access to the Fednet is controlled by the Fednet manager functionality. Similar to P2P file-sharing network, Fednets can have also an anonymous nature, in which the members do not need to know about other Fednet members.

The access to the resources, i.e. the second-level access control. In our survey, we encountered centralized and distributed access control to the shared resources. In grid networks that use grid map files, the CAS server, the VOMS server or the PERMIS policy engine the access control to the shared resources is centralized.

In the following cases the access control is distributed:

- In Fednets it is carried out at each PN by the PN agent;
- In grid networks that use AKENTI, it is carried out by each stakeholder;
- In grid networks that use PRIMA, it is carried out by each participating working group;
- In SVE it is carried out at each enclave by the enclave administrator;
- In P2P networks it is carried out by each peer.
- WCN, which is carried out at each Repeater AP and Wireless AP.

Complexity. The access control mechanisms and their complexity differ from technology-to-technology. The simplest mechanism is used in P2P networks, i.e. file encryption-decryption keys. The P2PWNC, FON and Grid map files use the mapping of the registered usernames to local accounts. Grid networks use access control policies based on privileges, capabilities or roles. The SVE uses role-based resource access control policies, defined by the enclave administrator. In our design of Fednets we use access control policies based on criteria such as the contribution of the PNs to the Fednet and the behavior of the PNs in the resource sharing.

Management scope. In PNs, the PN owner manages his/her own resources, while the SVE administrator manages the resources of the enclave that belongs to one organization. The domain agent in P2PWNC manages the access control to the resources of a Wireless Community Network with many users. In grids the scope is even larger, i.e. CAS and VOMS administrators manage the resources that belong to several organizations.

Being one of the most important issues in sharing resources, in this survey, we have focused on the access control mechanisms of group-oriented networking systems reported in the literature. Moreover, we discussed the advantages and disadvantages of each approach. We showed that although, the concept of sharing resources in these technologies that belong to different owners is similar to the concept of Fednets, the implementation of the access control mechanisms is different. Similar functionalities such as managing and controlling the group cooperation, the access control over the community resources are implemented differently. Our final remarks are the followings:

- There are different solutions for the access control in group-oriented communications. However, there is no universal solution. Each of the proposed solutions counters a particular problem and thus has its own tradeoffs. The access control can be carried out at the resource itself; in this case, the resources should have access control capabilities to interpret the user's attributes and make an authorization decision. This brings overhead and complexity at the resource side. Another solution is having a centralized entity, such as VOMS or CAS servers, who decide on behalf of the community how to grant the access to the resources. This approach releases the resources from extra task of controlling the access, and the complexity will move to a centralized entity. But this creates extra overhead for the whole system, since a centralized entity should be maintained. In addition, it is prone to a single point of failure.
- Fednets stand close to grids with a number of their characteristics. The scale of their geographic span, the number and the variety of resources and services shared between PNs, as well as the number of participants can vary based on the goal of the Fednet and type of its applications. Fednets are, in fact, a grid of personal networks cooperating in a P2P manner.
- Based on the survey we conclude that Fednets are distinguishable from the discussed related paradigms and technologies in the types of the participating resources (which are personal) and devices (which are mostly portable and battery powered). Fednets are enabled by the collaboration of individual Personal Networks, thus the administrative domains are PNs. The uniqueness of the Fednets among the existing related technologies is that Fednets are temporal, opportunity or purpose driven ad-hoc sharing of personal resources and services.

Fednets enable users to share their personal resources in a seamless, secure and flexible way. Fednets have a potential to cover a variety of P2P application categories, such as communication and collaboration (instant messaging), distributed computation (sharing available processing power), internet service support (sharing internet connection, multicasting services) and content distribution (digital media sharing). PNs and Fednets can be seen as a next generation networking concept that allows organizing personal devices in order to make them cooperate in an effective way.

References

[1] M. Ibrohimovna and S.M. Heemstra de Groot, "Sharing resources in group-oriented networks: Fednet and related paradigms" in Proceedings of UBICOMM 2008, Spain, Valencia, October 2008.

[2] I.G. Niemegeers and S.M. Heemstra de Groot, "Research Issues in Ad-Hoc Distributed Personal Networking", Kluwer International Journal of Wireless and Personal Communications, Vol. 26, No.2-3, 2003, pp.149-167.

[3] I.G. Niemegeers and S.M. Heemstra de Groot, "FEDNETS: Context-Aware Ad-Hoc Network Federations", Wireless Personal Communications Vol. 33, N.3-4, pp. 305-318, Springer 2005.

[4] I. Foster, C. Kesselman (editors). "The Grid: Blueprint for a New Computing Infrastructure". Morgan Kaufmann Publishers, USA, 1999.

[5] Wireless Community Networks, http://wcn.cnt.org, 12.05.2009.

[6] M. Ibrohimovna and S.M. Heemstra de Groot, "Proxy-based Fednets for sharing personal services in distributed environments," in Proceedings of ICWMC 2008, Greece, Athens, July 2008.

[7] IST 6FP MAGNET, http://www.telecom.ece.ntua.gr/magnet/, 12.05.2009.

[8] The Dutch Freeband Communications Project PNP2008, http://www.freeband.nl. 12.05.2009.

[9] I.Foster, C.Kesselman, S.Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", International J. Supercomputer Applications, 15(3), 2001.

[10] CrossGrid project. http://www.eu-crossgrid.org/, 12.05.2009.

[11] AccessGrid project. http://www.accessgrid.org/, 12.05.2009.

[12] Wilkinson, Grid Computing, Lecture notes. http://www.it.uom.gr/teaching/unc_charlottePPG/grid.htm, 12.05.2009.

[13] I. Foster, presentation "The Grid: Beyond the Hype," Argonne National Laboratory and University of Chicago, September 14th, 2004.

[14] L. Pearlman, V. Welch, I. Foster, C. Kesselman, S. Tuecke, “A Community Authorization Service for Group Collaboration”, in Proc. of the IEEE 3rd Int. Workshop on Policies for Distributed Systems and Networks, 2002.

[15] R. Alfieri, R. Cecchini, V. Ciaschini, L. dell’Agnello, A. Frohner, A. Gianoli, K. L˝orentey, and F. Spataro. Voms: An authorization system for virtual organizations. Proc. of the 1st European Across Grids Conference, Santiago de Compostela, February 2003.

[16] D. Chadwick and O. Otenko. The permis x.509 role based privilege management infrastructure. The 7th ACM Symp.on Access Control Models and Technologies, 2002.

[17] M. R. Thompson, A. Essiari, and S. Mudumbai. Certificate-based authorization policy in a pki environment. ACM Trans. Information Systems. Security, 6(4):566–588, 2003.

[18] M. Lorch and D. G. Kafura. The prima grid authorization system. Journal on Grid Computing, 2(3):279–298, 2004.

[19] D. Shands, R. Yee, J. Jacobs and E.J.Sebes, “Secure Virtual Enclaves: Supporting Coalition Use of Distributed Application Technologies”, in Proc. of NDSS 2000, San Diego, California, February 2000.

[20] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu. Plutus: scalable secure file sharing on untrusted storage, Proc. of FAST ’03:2nd USENIX Conference on File and Storage Technologies, San Francisco, CA, USA 2003.

[21] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong. Freenet: Distributed Anonymous Information Storage and Retrieval System. Lecture Notes in Computer Science, 2009:46, 2001

[22] E. C. Efstathiou at al., “Stimulating Participation in Wireless Community Networks”, in Proc. of IEEE INFOCOM 2006, Barcelona, Spain, April 2006.

[23] FON. www.fon.com, 12.05.2009.

[24] J. Vollbrecht, P. Calhoun, S. Farrell, L. Gommans, G. Gross, B. de Bruijn, C. de Laat, M. Holdrege and D. Spence, IETF RFC 2904, AAA Authorization Framework, August 2000.

[25] R. Yavatkar, D. Pendarakis, R. Guerin, A Framework for Policy-based Admission Control, IETF RFC 2753, January 2000.

[26] L. Pearlman, C. Kesselman, V. Welch, I. Foster and S. Tuecke, The community authorization service: status and future, CHEP03. March 24-28, 2003, La Jolla, California.

[27] The DataGrid Project. http://www.edg.org/, 12.05.2009.

[28] S. Tuecke, V. Welch, D. Engert, L. Pearlman, and M. Thompson. Internet X.509 Public Key Infrastructure Proxy Certificate Profile. RFC 3820, June 2004.

[29] M. Lorch, D. Adams, D. Kafura, M. Koneni, A. Rathi, and S. Shah. The prima system for privilege management, authorization and enforcement in grid environments. In Proceedings of the 4th Int. Workshop on Grid Computing - Grid 2003, Phoenix, AZ, USA, November 2003.

[30] V. Welch, I. Foster, C. Kesselman, O. Mulmo, L. Pearlman, S. Tuecke, J. Gawor, S. Meder, and F. Siebenlist. X.509 proxy certificates for dynamic delegation. In 3rd Annual PKI R&D Workshop, April 2004.

[31] David Chadwick, Sassa Otenko, Von Welch, Using SAML to Link the Globus Toolkit to the Permis Authorisation Infrastructure, Springer Boston ISSN 1571-5736 (Print) 1861-2288 (Online) Volume Volume 175/2005.

[32] Shibboleth. A Project of the Internet2 Middleware Initiative. http://shibboleth.internet2.edu/, 12.05.2009.

[33] C. Neuman, S. Hartman, K. Raeburn, RFC 4120, The Kerberos Network Authentication Service (V5), July 2005.

[34] Grid Security Infrastructure, GSI. http://www.globus.org/security/overview.html, 12.05.2009.

[35] J. Novotny, S. Tuecke, and V. Welch. An online credential repository for the grid: MyProxy. In Symposium on High Performance Distributed Computing, San Francisco, August 2001.

[36] Security assertion markup language. http://saml.xml.org/saml-specifications, 12.05.2009.

[37] S. Chokhani and W. Ford, Internet X.509 Public Key Infrastructure. Certificate Policy and Certification Practices Framework. RFC 2527, March 1999.

Case-based Decision Support for the Assessment of Bridges

Bernhard Freudenthaler, Reinhard Stumptner,
Josef Küng
FAW – Institute for Applied Knowledge Processing
Johannes Kepler University
Linz, Austria
e-mail: {bfreudenthaler, rstumptner, jkueng}@faw.at

Georg Gutenbrunner

VCE Holding GmbH
Vienna Consulting Engineers
Vienna, Austria
e-mail: gutenbrunner@vce.at

Abstract—**The main idea of this contribution is computer aid for Structural Health Monitoring activities. The increasing age of infrastructure makes actions necessary to predict lifetime and to guaranty safety, especially for critical structures like bridges for example. Thereby, acceleration sensors are widely-used to measure (ambient) vibrations of structures, which are stored by a connected computer system for later processing. These records are a basis for following procedures and make an assessment of a building's condition possible. Due to several reasons, the process of analysing a signal is very complex in particular because of the individuality of each structure. This means that the measurement results (characteristics of a building) strongly depend on structure-design and a marginally different design (from layman's point of view) can cause completely different measurement results. Consequently, only an experienced expert can interpret a measurement correctly and still, this analysis process is difficult and time-consuming, what necessitates computer aid for the interpretation to speed it up and to improve the quality of results. This is the point where decision support in terms of Case-based Reasoning can be introduced. The idea is to transfer the expert's experience (description of the structures' designs and measurements incl. interpretation) into a so-called case base which is continuously growing and enhanced by future experience. The Decision Support System can, relying on these cases, compare new measurements of possibly unknown structures with measurements of known buildings (from the case base) and suggest an interpretation by means of adapting past interpretations, which were taken under similar conditions (similar structure design) using certain similarity measures.**

Keywords: Bridge Monitoring; Case-based Reasoning; Decision Support System; Structural Health Monitoring

I. INTRODUCTION

Bridges play a major role in the higher transportation infrastructure. They represent large and expensive civil engineering structures with great importance to our economy and society and are, moreover, often exposed to extreme environmental and meteorological conditions. In order to deal with problems caused by these possible influences, intelligent solutions are needed. Fortunately, engineering and monitoring can ensure the bearing capacity of bridges to resist these conditions and thus, negative impacts on our economy and society can be prevented.

When bridges reach the end of their service life, which can be the result of structural damage and/or material degradation, they should have reached a minimum acceptable performance level. For the determination of this level many significant factors have to be taken into account. A Bridge Management System (BMS) can evaluate the adequate time for improvements on a bridge and it can improve the overall condition of an agency's network of bridges right in time.

A BMS is a decision support tool that consists of the following three major parts:

- Inventory (data regarding the characteristics and condition of the bridge),
- Inspection (examinations of the bridge) and
- Recommendations (regarding the maintenance and improvement of the bridge).

As an additional and actually very important feature, a BMS is also capable of prioritizing the allocation of funds. Therefore, BMSs are important for every stage of a bridge's life.

The importance of the current topic comes even clearer, when considering that the global higher transportation network operates about 2.5 million bridges. Current BMSs categorise these bridges with various methodologies and approaches. This results in very inhomogeneous figures. In 2005 the U.S. Federal Highway Agency (FHWA) stated that 28% of their 595,000 bridges are rated deficiently. Only a portion of it (about 15%) has structural reasons. In Europe this figure varies around 10%, whereas for Asian networks no such figures are available. Nevertheless, if we consider an average of 10% deficiency, we look at 250,000 bridges that definitely require structural health diagnostic, improvement and monitoring. Structural Health Monitoring (SHM) shall also be used preventively before bridges become deficient. This considerably enlarges the number of applications of Bridge Monitoring [21].

SHM is the implementation of a damage identification strategy to the civil engineering infrastructure. Damage is defined as changes to the material and/or geometric properties of these systems, including changes to the boundary conditions and system connectivity. Damage affects the current or future performance of these systems.

The damage identification process generally is structured into four levels [19]:

- Damage detection, where the presence of damage is identified,
- Damage location, where the location of the damage is determined,
- Damage typification, where the type of damage is determined and
- Damage extent, where the severity of damage is assessed.

Extensive literature on SHM has been developed over the last 20 years [8]. This field has matured to a point where several broadly accepted principles have emerged. Nevertheless, these principles are still challenged and further developed by various groups of interests. The strategies in mechanical engineering or aerospace take different approaches. However, the civil engineering community can considerably benefit from these efforts.

At the Stanford SHM workshop in 2005 Farrar and Worden [10] specified axioms for Structural Health Monitoring, which are an attempt to formulate common rules and understanding to support the "fundamental truth" that has been argued by the community. These axioms do not represent operators for SHM. In order to generate methodologies, it will be necessary to add a group of algorithms, which carry the SHM practitioner from data to a decision. The discipline of statistical pattern recognition is proposed for this approach. The axioms formulated are:

- Axiom 1: The assessment of damage requires a comparison between certain system states.
- Axiom 2: The existence and location of damage can be identified in an unsupervised learning mode, but the type of damages and damage severity can only be identified in a supervised learning mode.
- Axiom 3: Intelligent feature extraction is necessary because the more sensitive a measurement is to damage, the more sensitive it is to operational and environmental changes which do not have to be classified as damages.
- Axiom 4: There is a trade-off between the sensitivity of an algorithm to damages and its sensitivity to noise.
- Axiom 5: The size of a damage that can be detected from changes in the system dynamics is inversely proportional to the frequency range of an excitation.

The information of greatest interest is the knowledge about the condition of a bridge or its single elements. SHM provides the opportunity to quantify the condition and to provide the basis for decisions. Fortunately, due to bridges' importance for economy and society as well as their high vulnerability, procedures and tools of SHM may be best developed for them.

This contribution illustrates the possibilities of (semi-) automatic assessment in the field of Structural Health Monitoring, whereas at first the related research is discussed. The next chapter (Motivation for Bridge Monitoring) shows the requirements of the industry to support the conventional evaluation of measurement data by an intelligent system. For this purpose, an introduction to Decision Support Systems and Case-based Reasoning is provided. Finally, the current state of a research prototype for the Case-based Decision Support System for Bridge Monitoring and intended future work is shown.

II. RELATED RESEARCH

Due to the constant aging of our infrastructure, the field of Structural Health Monitoring has attracted a great deal of attention. In order to reduce the costs for maintenance and to increase the safety level of structures, the request of structural reliability, evaluation and remaining lifetime assessment is assuming a major importance. The use of non-destructive dynamic testing methods for the evaluation of the structural performances provided important steps forward. On the one hand, an improvement of the data reliability could have been gained; on the other hand, it succeeded in overcoming the limitations of traditional visual inspection methods.

In a next step, the measured response of a structure can be used in conjunction with several different numerical methods. These methods can mainly be divided into two categories: model-based and parameter-based. The first one is relying on a reference model, e.g., Finite Element (FE) model. By contrast, the main characteristic of the second one is a general mathematical description of specified parameters or system features, e.g., analysis of time series details by means of wavelet theory.

System identification with respect to determination of damages usually is done by extracting normal modes and frequencies. Based on them, engineers are able to calculate stiffness and damping coefficients. By updating initial mathematical models with finite element methods to predict the expected values of the actual measurements, damages can be localized and quantized as well. A comparison with reference data from earlier measurements allows experts to make statements on the structure's safety and furthermore gives the possibility to make lifetime predictions for the investigated bridge. However, Structural Health Monitoring produces a flood of data and the fact that each bridge - actually any arbitrary structure - has different dynamic parameters makes a manual analysis and interpretation very time-consuming and expensive.

Another main disadvantage of manual analysis is the subjective interpretation of human experts. Each expert interprets a measurement differently, based on his level of experience. Therefore, there is a strong need for intelligent Decision Support Systems (DSS) in safety assessment and lifetime prediction for civil engineering structures in general, and for bridges in particular. Case-based Reasoning (CBR) seems to be an appropriate approach to work with huge measurement data packets. For supporting engineers in interpreting measurement results and in making decisions, reasonable case sensitive DSSs have to be developed and adapted. CBR systems have a powerful cyclic problem solving core process. Based on known similar cases (stored past problems and their solutions), CBR helps engineers in interpreting certain situations. Since the main objective of CBR is not to develop new solutions for new problems but to reuse known problems and solutions, the reasoning process is comparably fast. CBR can be used for example to interpret measuring data of periodic measurements or as an integrated alert system for permanently monitored civil engineering structures.

A very common model for Decision Support Systems is the phase model by Simon [20]. It consists of three phases [15][12], namely

- Intelligence,
- Design and
- Choice.

The phase “Intelligence” is responsible for recognising problems. The indicators are very often weak, so that they have to be identified in advance. Hence, the engineer realises divergences of the norm and recognises the existence of a problem.

The phase “Design” serves decision makers as a platform to find alternative solutions for recognised problems by using already known solutions of similar problems.

Finally, the phase “Choice” allows decision makers to define and set criteria for finding new alternative solutions, which actually might perform better. Solutions for the actual problem can either be found by using known alternatives or new ones can be created. The objective is to select one single solution for the problem. The support for the decision-making process in this phase is exactly the typical application area of Decision Support Systems.

A. *Ambient Vibration Monitoring*

Each structure has its typical dynamic behaviour, which may be interpreted as “Vibrational Signature”. Changes in a structure, such as all kinds of damages are leading to a decrease of the load-carrying capacity and have effects on the dynamic response. This fact implicates to use the measurement and monitoring of the dynamic response characteristics for evaluation of the structural integrity.

Different types of bridge vibration tests exist: the bridge can either be excited with a heavy shaker or drop weight (Forced Vibration Testing) or by ambient excitation such as wind, traffic and micro seismic activity (Ambient Vibration Testing). The latter (Ambient Vibration Monitoring) is the fundamental principle used in the BRIMOS® technology (Bridge Monitoring System) which has been used in the field of Structural Health Monitoring for many years. It has the big advantage that no expensive equipment is needed to excite the bridge and that the traffic does not have to be interrupted.

The term Structural Health Monitoring in the meaning of Ambient Vibration Monitoring comprises the recording of the dynamic behaviour by the use of measuring instruments as well as the evaluation and analysis of the measured signals. The fundamental tools of health monitoring are system identification, damage determination and localization as well as safety assessment and the maintenance management for infrastructure.

The analysis provides the determination of the modal parameters, namely the structure’s natural frequencies, its mode shapes and its damping coefficients. These parameters, which are gained from the measurements, represent the real condition of a structure and are used to update mathematical models of a structure or are simply compared to reference data from earlier measurements.

Up to now, the analysis of measurement data requires the knowledge of an expert. This forms a weak point in the whole procedure, since the work done by experts is time-consuming and results are subjective. Therefore, a system supporting the engineer who interprets measuring data would be desirable, in order to make analysis easier and faster.

B. *System Identification*

Unfortunately, calculation models for determining stresses and consequently for measuring structures only represent an approximation to reality and have to be calibrated. For the determination of the conformity between the calculation model and the actual load-bearing behaviour up to now frequent stress tests (for example at railway bridges) have been carried out and the measured deformations (flexures) were compared with calculated reference values. Based on this, conclusions can be drawn on the load-bearing safety and performance capability of the structure.

A simpler and by far better method for the determination of these parameters is based on the determination of the dynamic characteristic by ambient vibration measurements. With these measurements, the vibration behaviour of a structure is recorded, evaluated and interpreted under ambient influences, e.g., without artificial excitation, by means of highly sensitive acceleration sensors.

The methodology to make conclusions on the load-bearing capacity of a structure by measuring its dynamic behaviour and to check mathematical model assumptions already is very old. In [9] there is a report on stress tests between 1922 and 1945 in Switzerland where tests by free oscillations at the aerial Beromünster in 1941 are described. The results were used for checking the calculation assumptions, deviations between measured and calculated results were interpreted and statements for similar future towers were done.

The checking of structures by means of dynamic measuring methods has a long tradition in Switzerland. It was carried out until the beginning of the 1990s in the form of tests by free oscillations by means of initial strains or intermittent stresses and by excitation with unbalance exciters or hydraulic shakers. Similar tests were also carried out in Austria and Germany for scientific purposes but at a much smaller scope. However, they were not extensively applied for system identification or check and calibration of calculation models. In [5] it is suggested to further develop dynamic procedures for the assessment of the maintenance condition of structures.

The rapid development of measuring technology on the one hand and computer technology as well as software on the other enables us to carry out dynamic measurements of ambient structure vibrations and their evaluation very quickly and with relatively low expenditures today.

Vibrations influencing the structure, which are due to natural excitation sources like micro-seismic phenomena, wind, waves etc., are regarded as ambient causes. The measuring and evaluation system BRIMOS® takes advantage of these progresses and opens a wide field of application to technology.

The dynamic characteristic of a structure can not only be used for a single check of calculation models. Furthermore, statements on the chronological development of the load-bearing capacity and therefore estimations on the remaining service life duration are enabled by measurements at certain intervals. Measurements at any moment supply snapshots of structural integrity and can be used in combination with parallel mathematical analyses for the determination of possible damages to the structure.

The list of decisive dynamic parameters to be determined for system identification is quite long and consists of eigenfrequencies, mode shapes (an example is shown in Figure 1), damping coefficients, vibration intensities, etc. During monitoring all analyses of system identification are applied.

In addition to the procedures of structure mechanics and dynamics, statistical methods have to be used which determine trends from large data quantities. The use of so-called trend cards, which clearly represent an eventual change of individual parameters by means of a time-frequency diagram, has proved successfully.

The eigenfrequencies are an essential parameter for the description of the vibration behaviour of a structure in the linear elastic field. A mode shape like in Figure 1 is a vibration form in which the structure oscillates with the respective eigenfrequency. The actual oscillation of a real structure is composed of the respective shares of the individual mode shapes.

The mathematical modal analysis provides both, the eigenfrequencies and the mode shapes of a structure, whereas in experimental modal analysis the eigenfrequencies are obtained as well and the mode shapes can be determined point by point (at the measuring points). Both methods have to be carried out for system identification. The actual static system is obtained by comparing the measuring results with the calculated values and by adaptation of the calculation model to the measurements. In order to get a correct image of the actual load-bearing system, one must not restrict oneself to the first eigenfrequency and the respective modal form. In fact, the consideration of several, also higher frequencies and the respective forms is required.

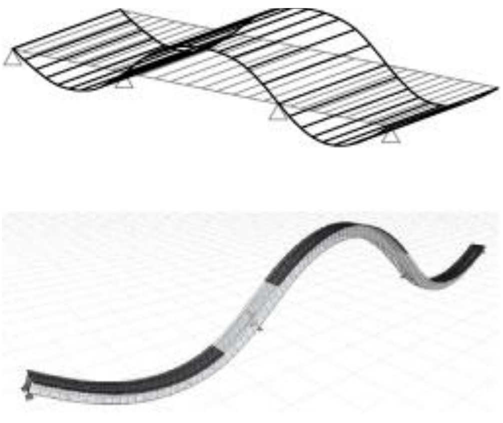


Figure 1. First Modeshape of an Austrian Danube Bridge consisting of three Spans

C. BRIMOS®

BRIMOS® (Bridge Monitoring System) is an application for system identification and the detection of damages in bridges as well as any other civil engineering structure. Its development is based on several research projects started almost 15 years ago. About 1000 structures have been assessed so far and the experience has been incorporated into the assessment procedure. It is based on the already mentioned “Vibrational Signature” of a structure, which is obtained by a measurement campaign. Depending on the extent of this campaign various properties can be computed, which are combined to the BRIMOS rating. This classification allows a fast identification on the structure’s integrity as well as the corresponding risk level. The results are based on

- Measured dynamic parameters (like eigenfrequencies, mode shapes, damping pattern in the lengthwise direction, vibration intensity and static as well as dynamic vertical displacements),
- Visual inspection,
- Finite Element model-update and
- Reference data (BRIMOS-Database and BRIMOS Knowledgebase).

The result is a factor, which relates to a predefined risk level.

D. Bridge Monitoring

The extent of monitoring is mainly depending on required results. Currently five levels are used in order to determine the depth of investigation [21]:

- Level 1: Rating
  It represents the conventional assessment of the structure starting with a visual field inspection that provides a subjective impression of the condition of a structure. Some preliminary analytical investigation is performed in order to provide a rating as a basis for decisions. This would be a typical application of a bridge management system like PONTIS or DANBRO. Many bridge owners use certain databases to store the results.
- Level 2: Condition Assessment
  A rough visual field inspection has to be an element of any SHM campaign. Afterwards a decision has to be reached whether the conventional approach is satisfactory or an extended or even sophisticated

additional approach has to be considered. This determines the type and quantity of instrumentation. For condition assessment a simple instrumentation is sufficient and a simple Decision Support System would provide the necessary additional information. Storage and pre-processing of data should be done in the existing database where a link to existing conventional tools is available. The monitoring can be performed at single spots only.

- Level 3: Performance Assessment
  This intermediate level uses the same procedure as described in level 2. The level of assessment and performance elaboration in the decision support process is considerably higher since additional information like mode shapes is measured and determined. This provides additional indicators for the assessment and will illustrate the performance of a structure.
- Level 4: Detail Assessment and Rating
  The next step is to establish an analytical model representing the structure and the model is compared with the monitoring results. In case that phenomena are detected that cannot be explained based on the records, further steps have to be taken to clarify the situation. The most obvious method is to introduce a permanent record over a certain period of time to capture the necessary phenomena being responsible for the specific case. Load testing also has been proven successfully to establish performance parameters. With these results a simple model update can be performed to assess the results and provide a rating. Certainly, extensive monitoring is required. The records shall cover at least 24 hours, but shall rather be much longer to capture environmental aspects and traffic situations as completely as possible.
- Level 5: Lifetime Prediction
  For a serious lifetime prediction, the records taken have to be long enough to cover at least three cycles relevant for the structure. This normally is in an interval of three years. Simulation should be run on the analytical model in order to achieve a theoretical performance for comparison. To handle the major quantity of data, software for decision support is required. Load testing would be done targeted and extensive. In addition, micro structural testing might be useful in order to look into the performance of single elements of a structure. The update process would be extensive and considering several conditions of a structure. This in particular includes the loaded and unloaded case and all the nonlinearities involved. The monitoring system shall be operated online, probably web-based, providing a warning in case of critical/unknown situations. The final lifetime prediction could be performed.

The costs related to these procedures are mainly depending on the extent of the monitoring campaign and the number of man-hours to be invested in modelling, simulation and update procedures. The effort can also be influenced by the type of a structure (e.g., number of spans). For the future of Structural Health Monitoring it is expected that the monitoring-costs will be rather reduced than increased. This can happen through the introduction of time saving modelling procedures and sophisticated monitoring software [21].

III. Motivation for bridge monitoring

Bridge Monitoring (BM) has undergone a long development period and many useful results have been produced. Nevertheless, the transformation into a business case has been scarcely managed by 2008. The three main reasons for that are:

- BM is a very complex issue. The key players concentrate on issues which the ordinary bridge owner is not interested in. A joint language has not been found and appealing method statements are lacking.
- The discrepancy between the expectations of the owners and the services that can be provided by the available budget is huge. The community has not been able to explain that the new methods do not eliminate the problem of aging or damaged bridges but can only serve it in a better way. For this reason, monitoring campaigns, which are so expensive that they can only be of scientific interest, mostly are performed in the frame of research projects.
- The involved hardware is still very expensive and not robust. The discrepancy of life expectation of a typical bridge with 100 years and three years for a monitoring system is unacceptable.

Apart from these facts there are other aspects which are related to the national practice of bridge management and cultural differences.

Three main driving forces have been identified that enable the performance of a reasonable BM campaign:

- Responsibility: Meaning the existence of a standard or recommendation that obliges the owners to monitor their structures.
- Economy: If BM can prove that it saves money.
- Curiosity: Owners of bridges are very often willing to spend money on creating better knowledge of their bridge stock, especially when there has been reasonable doubt of an actual condition.

The monitoring community has managed to issue some guidelines and recommendations. The latter form the basis for eventual orders. Nevertheless, they cannot be seen as an obligation to apply Bridge Monitoring.

A good conception has been promoted and implemented in Austria. The regulations for bridge management allow both, the visual inspection and the monitoring campaign. In case that monitoring enables to achieve better quantified results, the inspection period can be increased up to 100%.

This saves money on inspections which can be invested in monitoring. A better service is performed at the same costs.

A. State of the Art

The selection of a suitable observation concept has to be mainly based on external factors. These are the number of structures to be observed in combination with the budget available. For this purpose it is necessary to offer services on increasing quality levels. The levels can be subdivided into spot, periodic, permanent and online assessment campaigns at structures [13]. The respective features are:

- A spot observation shall comprise a very quick measurement campaign with only few sensors which can be simply handled. It shall provide information on the general condition of a structure in order to create a ranking.
- Periodic assessment means a measurement campaign on a structure which is repeated after a specified period of time, to generate information on the performance over time. The single spot information might comprise rather long periods.
- Permanent observation and assessment of structures becomes necessary when certain limits are passed. This observation allows a very detailed assessment based on permanent recordings and can help to implement quick decision making.
- Online observation and assessment allows warning through electronic media, either by SMS (Short Message Service) in the simple case or by online status through the internet. Decisions might be taken by the computer based on the measurement data. These alert systems would only be applied at extremely critical structures.

In general, it has to be stated that clients need and desire support of their work and not to create issues that make it more complicated. In respect to that the procedures have to be carefully watched and permanently improved. The information policy also plays a major role in the client-consultant relationship. The new methodologies are rather complex and require a deep understanding of structural dynamics, physics and measurement techniques. Due to the fact that this expertise is rarely available at the owners engineering department, the fear to be exposed to unknown black box applications has to be taken from them with bringing transparency to the systems.

Nevertheless, they spend considerable amounts of money on monitoring actions and would like to be informed frequently about progress and results. Therefore, it has to be ensured that the technology-part is in good and competent hands and that they will receive the information they desire. From a historical point of view the best success has been achieved with very simple reporting techniques. A periodic report received by e-mail comprising single page information generally turned out to be preferred.

The main information is provided in a single window, where upper and lower normalized thresholds are given and measurement results of this period are placed within these thresholds. With a single look at this graph, the personnel can see whether any of the thresholds has been exceeded at once. When all indicators are green, the client can be pacified and knows that the ordered observation is permanently working.

The periodic report mentioned above provides the following information:

- A photo and a system plot of the structure of interest for an easy and quick identification.
- A window where the periodic results are placed within the relevant thresholds over the observation period.
- Eventually a second window, containing special information required by the client, such as wind speed information or any other desired quantity.
- Finally, a rating should be provided, based on the measurements taken in the reporting period. This should enable the client to immediately see whether any changes have happened.
- Eventually, the specification of a remaining life capacity can be provided if the necessary data are recorded.

Besides this one-page record for the client, also a scientific report for the expert is generated by the DSS. This makes a quick assessment of all the single measurements possible in order to create expertises or to learn from operation. On average, the system is calibrated with the information gained over a certain period of time. This might also comprise a change in the rating and would update the remaining life capacity based on existing knowledge.

B. System Requirements

During the last decades, the capability of both, computers and sensors, have undergone an explosive growth presenting many new challenges of how to manage the resulting amount of data. Scientists have often found themselves confronted by gigabytes of complex data that contained comparatively little information of actual interest making successful management almost impossible.

The problem of searching for the right information is very difficult even if one precisely knows where it can be found. However, it gets almost insoluble if the location additionally is not known exactly. Due to the complexity of data itself and the “human error rate”, two phenomena can be recognized:

- Useful information is often overlooked, which leads to a poor utilization of data.
- Possible benefits of increasing data-gathering capabilities are only partially used.

Since humans have not undergone similar developments in measurement data management as has the technology behind the measurements themselves, one has to look for intelligent ways that help to solve this dilemma. Since manual data analysis is quite tedious and impractical, other concepts like computational tools and techniques for an automated analysis of large complex data sets have to be developed.

Furthermore, the evaluation of data and the assessment of structures must be carried out by experts with many years of experience at the moment. By developing a Decision Support System, the expert should not be replaced but essentially supported in his activities. Such a system should support the whole data process, from the receipt of data, preliminary sorting, filing, evaluation, assessment up to visualization of explored results. For the preparation of such a system it is required to establish something like a knowledge database. In the latter the criteria normally used by the expert for assessment could be mathematically formulated (= formulation of rules, knowledge acquisition). The advantage is that the knowledge basis can be continuously expanded and that no "forgetting" exists. By the use of several methods of statistics up to now unknown connections could be filtered out of the existing measurement data pool. It should be possible to integrate measurement data from third persons from different measurement systems. The whole system should be based on methodologies like being implemented by the BRIMOS software and include very recent approaches (fuzzy logic, neural networks in damage identification). For visualization of results a GIS (Geographic Information System) interface can be provided.

The backbone of such a system would be a huge database containing measurement data and corresponding results of the analysis from the past. This knowledge base is filled with material from past measurements up to present ones and consequently will grow continuously. Using such methodology, the system for the user may become a "living system" and may increase his/her trust. With every improvement the results are likely to become better and the thresholds might vary too. Thresholds should not be treated totally inflexibly and have to be adapted to new knowledge.

Finally, engineers would be interested in a system, where they can search for similar objects or similar measurement data for arbitrary objects and measurement data respectively. Then, by having the corresponding results of comparable measurements from the database, it might become possible to draw new conclusions for certain objects. This would represent a very interesting feature for periodic measurements but also for spot observations. Treating the database of the past measurements as a case base, by means of CBR and similar techniques, benefit might be expected for interpreting the measurement data of periodic measurements, spot observations and in particular of permanently monitored structures.

The annual expenses for bridge maintenance in Austria amount to approximately 130 million Euro. According to current predictions this value will triple in the next 15 years. New methods for structural assessment are required in order to identify urgently necessary measures and to reduce the expenditures for purely precautionary maintenance. The Decision Support System proposed in this contribution is to lower the costs and the time consumption for the evaluation of measurement data and condition assessment, and at the same time accuracy and objectivity shall be increased.

IV. DECISION SUPPORT SYSTEMS

Since 1950 [15] there were efforts to develop systems to support experts of various fields in taking decisions. Decision Support Systems represent an approach that tries to integrate many different disciplines into the field of computer science. A reason for the arise of DSSs was a wish to have a system helping humans to manage very complex situations. Soon these systems became more and more attractive for users and researchers.

Problem solving and decision making are very important tasks in all intelligent activities. One who makes decisions usually evaluates and chooses among different alternative decisions. Problem solving on the other hand is a task to find a "way" between a desired goal and what is given at the beginning, to find intelligent steps to reach this goal. Expert systems and artificial intelligence in general deal with this problem in the following way.

The stages of problem solving are:

- To recognise situations which call for actions,
- To formulate problems,
- To find actions and to set up goals,
- To evaluate and
- To choose.

There are two scientific approaches in this case:

1. The normative approach: Prescribe optimal behaviour, how decisions should be taken.
2. The descriptive approach: Understand how humans behave when they are solving problems and take decisions.

The normative approach was first developed in economics; "the rational economic man" is an important term in this connection. Later it was implemented in the fields of Operations Research and management science. The theory is based on a rationality paradigm; rational behaviour is prescribed by formal axioms. Normative models for decision making are called "formal models". The decision maker in the process of finding a decision calculates the consequences for each alternative decision, rates the results and tries to compute the optimal way to his/her goal. To do so, future consequences of current actions have to be predicted and suggestions have to be made. A DSS should make decision making more effective and implies a normative perspective of the problem. These normative theories help to analyse the structure of a decision.

The normative theories enable us to optimise decisions under certain conditions, as long as the problems do not get too complex. The complexity of real-world decision making mostly overtaxes this approach, see [15].

DSSs are useful especially when there is a fixed goal but no algorithmic solution. The paths of solution mostly are very numerous and user-dependant. This leads to the main goal of DSSs, namely to improve decisions by better understanding and preparation of tasks which lead towards evaluation and choosing. Ill-structured problems in this regard are processes where there is no known or clear method to reach a solution, because the nature of the problem is complex and unclear or there arise situations which are new and consequently unknown. Well structured problems can be seen as decision making processes which are routine and repetitive. Usually it is not possible to fully automatise information processing to reach a conclusion. Only if an information processing task can be mapped to an algorithm then the decision process is structured and it can be implemented in a computer program to reach an automated solution.

Decision support for "unstructuredness" is accommodated in:

- The nature of requests made on a DSS.
- The manner in which a DSS responses are utilised.
- The recognition of alternative methods for satisfying a request.

Structured problems are routine because they are unambiguous, as there is a single solution method. If problems become less structured, then there exists an increasing number of alternative solution methods whereby solutions may not be equivalent. A completely unstructured problem in contrast has unknown solution methods or solutions are too numerous to evaluate.

Mainly in the field of management there are many situations where decisions have to be taken for non-programmable problems. The development of Artificial Intelligence technology especially in such connection has enlarged the spectrum of application of DSSs.

A. *Decision Support Systems for Real-World Problems*

Computer Systems nowadays are frequently used to solve numerous "real-world problems". In 1958 for example a computer system was used to find the optimal allocation of water between Egypt and the Sudan. The water system contained five major dams, several other barrages and control points and the monthly volumes of inflow between 1905 and 1942 were used as input data. The system was developed by IBM; it became the first considerable example for a computer assisted real-world plan [15]. In the 1960s the use of such devices was spreading rapidly.

Due to improved computer hardware and because of a changed attitude of users towards computers in general, computer applications to real-world problems became more and more attractive. Today, computers are used to collect, store and retrieve data, display and present it in different ways and help humans in understanding complex situations and problems. The computer became a "complete" information processor as part of complex information systems. Real-world computer systems process information such as digital values, analogue signals, images, etc. A computer with sufficient software may represent the physics or chemistry laboratory for scientists for instance. With using computer simulations for experiments costs, risks and time can be saved in many scientific fields. Consequently, there is no need to prove that an idea for an experiment will be useful. Using computers for experiments allows the planner or manager to make mistakes without consequences.

V. Case-based reasoning

According to Aamodt and Plaza [2], "Case-based reasoning is a recent approach to problem solving and learning (…)". Case-based Reasoning is a cyclic problem solving process, whereby already known knowledge is used to solve new problems. This knowledge is represented in form of cases which consist of a problem and a corresponding solution. The cases are stored in the so-called case base mainly providing the functionality to search for similar problems. Main objectives of CBR are the reuse of solutions of similar problems, no new problem solving processes whenever it is not necessary, no new solutions have to be developed for new problems if the case base contains a comparable problem and finally, the creation of solutions is rapid and cost-effective. Figure 2 shows one of the most important fundamentals of Case-based Reasoning, namely the CBR-cycle according to Aamodt and Plaza [2].

The cycle is subdivided into four phases: Retrieve, Reuse, Revise and Retain.

- Retrieve: Due to a new problem a new case is defined. Accordingly similar cases are retrieved from the case base where all known cases and general knowledge are stored. The retrieval of similar cases is operated by so-called similarity measures such as the similarity measure by Hamming, the Tversky-contrast model or even the Euclidean distance in an n-dimensional space.
- Reuse: If one has found one or more similar cases, the most similar retrieved case(s) is(are) combined with the new case, whereon the CBR-system can suggest solutions for the initial problem.

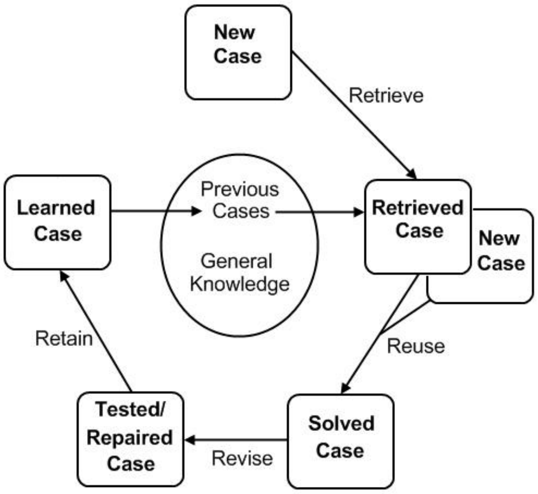


Figure 2. CBR-Cycle (According to [2])

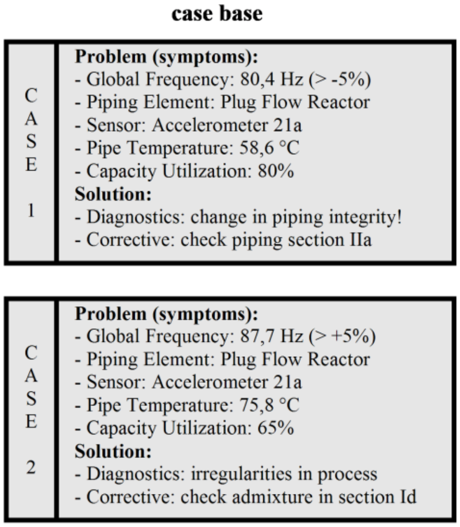
- Revise: The suggested solution is tested to demonstrate the ability of the CBR-system to solve the initial problem. If the retrieved solution is faulty, it can be adapted and a confirmed solution is created.
- Retain: Useful experience (significant cases) is stored for future reuse. The operator can add the new case or a learned case to the case base or the CBR-system creates and stores the new case automatically or semi-automatically. Finally, the case base grows and becomes more intelligent for future problems.

A. *Simple Example*

In this chapter we present a simple example to show how Case-based Reasoning could be used in the field of Structural Health Monitoring.

An example for a case base can be seen in Figure 3 where (in this very simple example) two cases are stored. A case consists of a problem (symptoms) with certain attributes and an appropriate solution (including diagnostics and corrective). When a new problem arises (shown in Figure 4) at first a solution is not available. To provide it, instead of starting a problem solving process, knowledge is reused from already known cases.

It has to be defined, which attributes or parameter values can be compared with each other. In this example, the attributes “Global frequency”, “Piping Element”, “Sensor”, “Pipe Temperature” and “Capacity Utilization” are used.



case base

CASE 1

Problem (symptoms):
- Global Frequency: 80,4 Hz (> -5%)
- Piping Element: Plug Flow Reactor
- Sensor: Accelerometer 21a
- Pipe Temperature: 58,6 °C
- Capacity Utilization: 80%

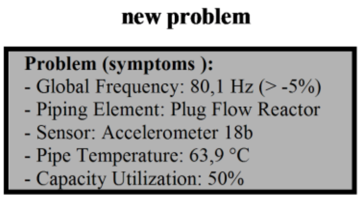
Solution:
- Diagnostics: change in piping integrity!
- Corrective: check piping section IIa

CASE 2

Problem (symptoms):
- Global Frequency: 87,7 Hz (> +5%)
- Piping Element: Plug Flow Reactor
- Sensor: Accelerometer 21a
- Pipe Temperature: 75,8 °C
- Capacity Utilization: 65%

Solution:
- Diagnostics: irregularities in process
- Corrective: check admixture in section Id

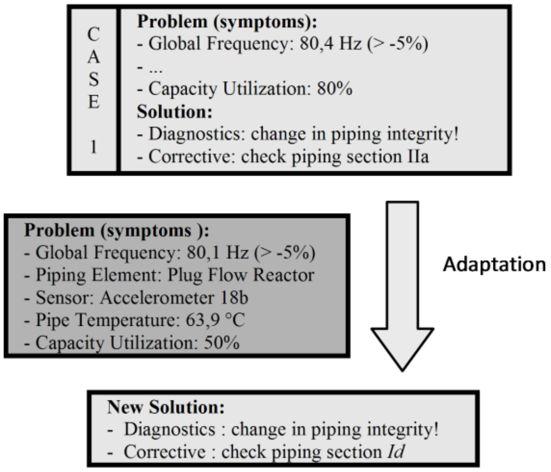
Figure 3. Simple Example – Case Base



new problem

Problem (symptoms):
- Global Frequency: 80,1 Hz (> -5%)
- Piping Element: Plug Flow Reactor
- Sensor: Accelerometer 18b
- Pipe Temperature: 63,9 °C
- Capacity Utilization: 50%

Figure 4. Simple Example – New Problem



CASE 1

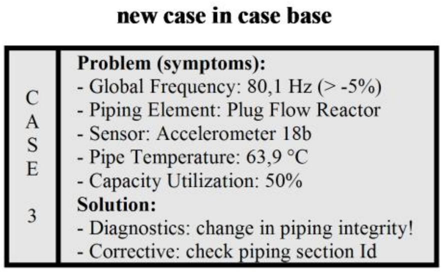
Problem (symptoms):
- Global Frequency: 80,4 Hz (> -5%)
- ...
- Capacity Utilization: 80%

Solution:
- Diagnostics: change in piping integrity!
- Corrective: check piping section IIa

Problem (symptoms):
- Global Frequency: 80,1 Hz (> -5%)
- Piping Element: Plug Flow Reactor
- Sensor: Accelerometer 18b
- Pipe Temperature: 63,9 °C
- Capacity Utilization: 50%

New Solution:
- Diagnostics : change in piping integrity!
- Corrective : check piping section *Id*

Figure 5. Simple Example – Adaptation



new case in case base

CASE 3

Problem (symptoms):
- Global Frequency: 80,1 Hz (> -5%)
- Piping Element: Plug Flow Reactor
- Sensor: Accelerometer 18b
- Pipe Temperature: 63,9 °C
- Capacity Utilization: 50%

Solution:
- Diagnostics: change in piping integrity!
- Corrective: check piping section Id

Figure 6. Simple Example – Result

One can compare the values of the attributes of new problems with the values of the attributes of cases in the case base. In this simple example no kinds of weights are used, for a real world system weighted attributes in general would be useful. If one compares the new problem to the cases in the case base, one can see that case 1 is more similar to the new problem than the case 2.

So, one can use the solution of case 1 and adapt it to the new problem. Figure 5 shows the adaptation of the reused solution to the new problem. The result shown in Figure 6 is a new case with the initial problem and the reused and adapted solution which can be stored in the case base. Thus, the case base grows continuously and probably becomes more intelligent for future problems.

B. *Similarity Measures*

Similarity measures play a great role for Case-based Reasoning. These measures are essential to be able to compare new problems with the cases in the case base. One can imagine that it is fundamental to choose the right methods of similarity measuring for given data. In the following, an example is shown to illustrate how one can calculate the similarity between cases. Therefore, the Generalized Similarity Measure by Hamming defined by the following formula (1) is used:

$$sim(x,y)=\frac{\sum_{i=1}^{n} w_i sim_i(x_i,y_i)}{\sum_{i=1}^{n} w_i} \qquad (1)$$

Table I shows a case base with five cases. The attributes again are “Global Frequency”, “Piping Element”, “Sensor”, “Pipe Temperature” and “Capacity Utilization” and each case x_1 ... x_5 has its individual parameter values.

When there is a new case y, one wants to know, which case in the case base is the most similar one to the new case y, see Table II.

To be able to use the Generalized Similarity Measure by Hamming, for each attribute functions have to be defined, e.g., how similar is a Plug Flow Reactor “PFR” to a Branch Connection “BC” or how similar is an Accelerometer “A15a” to an “A18b”? The functions e.g., can be defined like this:

- Global Frequency = $sim_{GF}(x_{GF},y_{GF})$ = For each hertz (Hz), which differs from case x_i to case y, the similarity value is reduced by 0,01.
- Piping Element = $sim_{PE}(x_{PE},y_{PE})$ =
    - Plug Flow Reactor: Similarity value of 1
    - Branch Connection: Similarity value of 0
- Sensor = $sim_S(x_S,y_S)$ =
    - Accelerometer 18b: Similarity value of 1
    - Accelerometer 15a: Similarity value of 0,75
    - Accelerometer 21a: Similarity value of 0,85
    - Accelerometer 24c: Similarity value of 0,5
- Pipe Temperature = $sim_{PT}(x_{PT},y_{PT})$ = For each degree Celsius (°C), which differs from case x_i to case y, the similarity value is reduced by 0,01.
- Capacity Utilization = $sim_{CU}(x_{CU},y_{CU})$ = For each percentage point (%), which differs from case x_i to case y, the similarity value is reduced by 0,01.

TABLE I. CASE BASE

Case	Attributes				
	Global Frequency (in Hz)	Piping Element	Sensor	Pipe Temp. (in °C)	Capacity Utilization (in %)
x_1	80,4	PFR	A 15a	58,6	80
x_2	87,7	PFR	A 24c	75,5	65
x_3	72,3	PFR	A 18b	78,4	53
x_4	92,7	BC	A 21a	71,9	90
x_5	78,4	BC	A 24c	85,1	50

TABLE II. NEW CASE Y

Case	Attributes				
	Global Frequency (in Hz)	Piping Element	Sensor	Pipe Temp. (in °C)	Capacity Utilization (in %)
y	80,1	PFR	A 18b	63,9	50

TABLE III. WEIGHTING COEFFICIENTS AND SIMILARITIES

Case	Attributes				
	Global Frequency (in Hz)	Piping Element	Sensor	Pipe Temp. (in °C)	Capacity Utilization (in %)
x_1	1	1	0,75	0,95	0,7
x_2	0,92	1	0,5	0,88	0,85
x_3	0.92	1	1	0,86	0,97
x_4	0,87	0	0,85	0,92	0,6
x_5	0,98	0	0,5	0,79	1
w_i	1	0,8	0,2	0,75	0,5

If one uses these functions for the similarity search, a new table with the similarities between the cases of the case base and the new case y can be generated, see Table III. The last row in Table III shows weighting coefficients (w_i), which represents the importance of an attribute for calculating the similarity.

To calculate the similarities between the cases in the case base and the new case y, the Generalized Similarity Measure by Hamming is used. The similarity between the case x_1 and the case y is shown in the following calculation (2):

$$sim(x_1,y)=\frac{(1*1+0{,}8*1+0{,}2*0{,}75+0{,}75*0{,}95+0{,}5*0{,}7)}{3{,}25} \approx 0{,}93 \qquad (2)$$

Using this formula for the other cases x_2 ... x_5, the following similarities can be calculated:

x1: 0,93
x2: 0,89
x3: 0,94
x4: 0,62
x5: 0,67

As one can see in this listing, the case x_3 is the most similar case to the new case y and would be used to find an already known solution for the new case y.

VI. CASE-BASED DECISION SUPPORT FOR BRIDGE MONITORING

The main idea of Case-based Decision Support for Bridge Monitoring is to support a human expert in the interpretation of measurement data taken from certain structures, especially from bridges. In general, the idea is to support the interpretation process by providing comparable measurements which may have lead to comparable interpretations. Thereby, in case of periodic measurements or spot observations, measurement results of similar structures should provide a basis for the interpretation of new measurements and in case of permanent monitoring, similar historical measurements can be taken into account to draw a conclusion to the state of the building. The ambient vibrations in the form of raw measurement data have to be interpreted by an engineer with profound technological knowledge and experience in the interpretation of such data. Another disadvantage of human interpretation, besides the time it consumes, is its subjectivity. Each expert interprets a structure differently. On the basis of these facts, a Decision Support System is needed to support engineers in interpreting measurement data and decision making in order to be able to take decisions faster and not to spend their time with doing rather easy routine work. Consequently, a main incentive of a Case-based Decision Support System for Bridge Monitoring is cost-effectiveness and promptness by finding solutions.

The development of the Decision Support System for the interpretation of periodic measurements is divided into the following steps, whereby for permanent monitoring step 1 in general is not necessary:

1. Search for Similar Bridges

First of all one has to search for similar structures to have a basis for further conclusions and comparisons. If the bridge, which should be observed, is already stored in the case base, then the engineer has to load its geometric data or has to feed the system with this information. The system now can compare this geometric information with the cases in the case base and search for similar bridges. Each kind of bridge (e.g., simply supported, continuous, cable stayed, suspended, etc.) has different attributes which have to be compared with each other. According to its nature, a bridge consists of different kinds of structural elements (e.g., bridge decks, cables, pylons, etc.) and for each structural element, different attributes are stored. For instance, a bridge deck among other things has attributes like length and breadth, main span, number of fields or material. The similarity between the cases in the case base in case of this system is the Euclidean Distance in an n-dimensional space. In case of two dimensions for instance the Euclidean Distance would be the “measurable” distance between two points. Using the following formula (3) the Euclidean (n-dimensional) space becomes a metric space. This fact makes the system become attractive for metric index structures to improve performance.

$$d(X,Y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \tag{3}$$

The system provides the most similar bridges with the distances to the analysed bridge. It is evident which bridge will be the most similar one in case that the observed bridge already is contained in the case base, namely the same one.

2. Search for Comparable Measurements (of Similar Bridges)

As a result of the first step (in case of periodic measurements), similar bridges due to their geometric data are retrieved. This is a preliminary selection in order to avoid the situation that different kinds of bridges, which accidently have similar measurement results, are compared which most likely would lead to a complete misinterpretation of the measurement. If the system only considers the measurement results of bridges and not the geometric data in a first step, disparate bridges (due to their geometric data) could have similar measurement results although they are totally different. In the second step, the already known measurement results of the retrieved similar bridges are provided. These results generally consist of modal parameters, namely the structure’s natural frequencies, its mode shapes and its damping coefficients which together represent the “real” condition of a bridge. In the following, the modal parameters of the similar bridges can be used as a suggestion for the interpretation of new measurement data of bridges which should be observed.

3. Support of Analysis Process

The steps for the interpretation and preprocessing of measurement data by a human expert can be stored in an adequate way in the case base. Thereby, similar bridges also have similar preprocessing steps for the interpretation of measurement results.

A. *Case-based Reasoning for Periodic and Permanent Monitoring*

As already mentioned in the previous chapters, there generally are two different applications of Structural Health Monitoring, namely periodic and permanent measuring. Depending on this kind of strategy the provision of decision support has to be adapted. While in case of permanent monitoring always one and the same structure is observed and decision support generally can rely on data from this certain structure, decision support for periodic or single monitoring on the other hand has to be handled differently. Decision support for structures which are measured periodically or ever for just a single time can only take into account measures of other structures to provide some kind of support. It is obvious that in this case only measures from similar structures can usefully contribute to the interpretation of such measuring data.

Below two activity diagrams are shown to illustrate the difference between periodic and permanent monitoring.

The first diagram represents activities for periodic monitoring and the second one for permanent monitoring.

Periodic Monitoring (shown in Figure 7):

- Measurement of a bridge
- Analysis of measured data
- Determination of similar bridges/cases by means of Case-based Reasoning
- Providing a suggestion about the condition of the bridge, based on the Case-based Reasoning system
- Accept or reject the suggestion

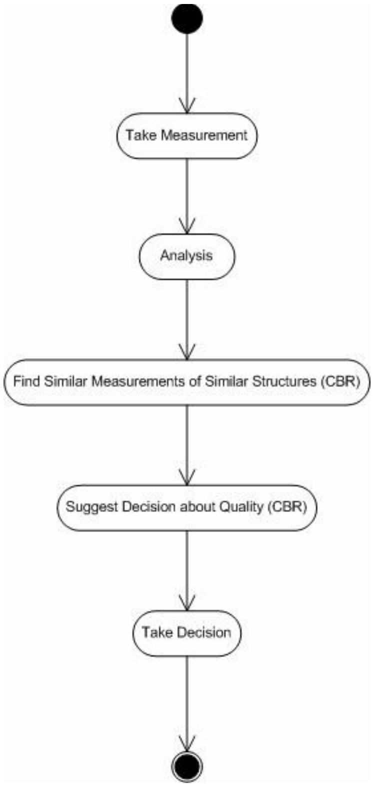

Take Measurement
Analysis
Find Similar Measurements of Similar Structures (CBR)
Suggest Decision about Quality (CBR)
Take Decision

Figure 7. Activity Diagram – Periodic Monitoring

Permanent Monitoring (shown in Figure 8):

- Measurement of a bridge (automatically)
- Generation of attributes for a representation as cases (preprocessing)
- Finding similar situations with the Case-based Reasoning system
- If the Case-based Reasoning system classifies the current situation as critical, then the system suggests possible solutions for solving this problem

The following chapters introduce more details like the integration of monitoring data, performance enhancement and the database model of the CBR system for example.

B. Data Preparation

After measurements of certain structures are taken, engineers start to analyse and classify them. For the analysis, measurement results (eigenfrequencies), and in most cases additional data, e.g., from visual inspections, are taken into account. Consequently, each case consists of a set of weighted attributes with different meanings and different data types. Due to the aim of representing the cases as points in a normalised n-dimensional metric space (each dimension has a finite range between 0 and 1), the definition of similarities/distances has to be well-thought-out.

The (Euclidean) distance between two cases X and Y is defined as following:

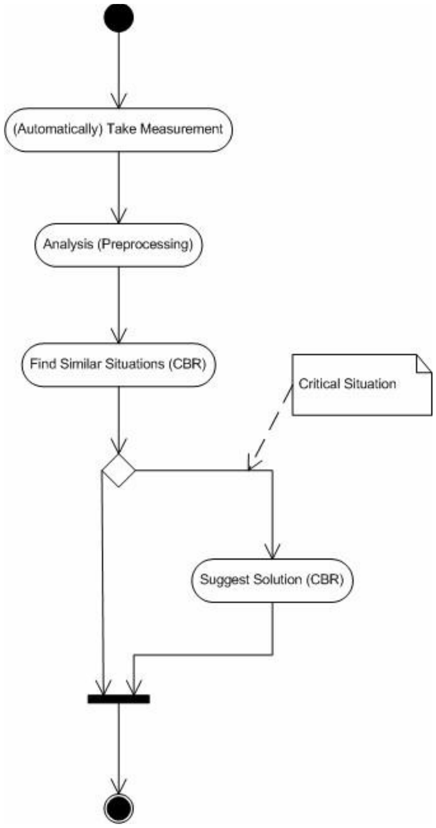


Figure 8. Activity Diagram – Permanent Monitoring

$$d(X,Y) = \sqrt{\sum_{i=1}^{n} f_i(x_i,y_i)^2} \tag{4}$$

The function f_i of the formula which is shown above returns a value between 0 and 1 representing the distance between two parameter values of a certain dimension. For the current application of the Case-based Reasoning system for Structural Health Monitoring it has turned out to be sufficient to rely on similarity representation in the metric space with numerical attributes (e.g., eigenfrequencies) on the one hand and predefined distances between parameter values on the other hand. Such distances for a certain attribute are defined in a matrix (which generally is symmetric: $d_{mn} = d_{nm}$) organised as following:

$$\begin{array}{cccccc} & \mathbf{v_1} & \mathbf{v_2} & \mathbf{v_3} & \mathbf{...} & \mathbf{v_n} \\ \mathbf{v_1} & 0 & d_{12} & d_{13} & ... & d_{1n} \\ \mathbf{v_2} & d_{21} & 0 & d_{23} & ... & d_{2n} \\ \mathbf{v_3} & d_{31} & d_{32} & 0 & ... & d_{3n} \\ \mathbf{...} & ... & ... & ... & ... & ... \\ \mathbf{v_n} & d_{n1} & d_{n2} & d_{n3} & ... & 0 \end{array}$$

Defining distances this way only is useful as long as all possible parameter values are known. As these distances in general have to be predefined by the user, the number of values has to be limited to a reasonable amount. In case of a symmetric distance matrix and a diagonal (d_{ii}) equal to zero, the number of predefined distances for a certain number of parameters n is $\frac{n^2-n}{2}$. Thus, for n = 20 the number of distances is 190. One has to realise that these dimensions soon will become unclear and unmanageable for a user.

f_i also can be a complex function, just having the constraint to return a value between 0 and 1 in our case. An attribute of the cases could be a graph for instance. There are algorithms to calculate the similarity between two graphs (Graph Edit Distance, see also [17][18]), so f_i would be this algorithm for distance calculation between attributes of the dimension concerned.

C. Data Model

Figure 9 shows the Data Model which gives an overview of the Case-based Decision Support System. There is a class called “CaseBase” where the name and a description of the case base are stored. The class “Case” is a container for the cases of the system. “AttributeBase” consists of all available attribute types with certain weights, which represent their importance. “SolutionBase” describes all available solution types. The parameter values of attributes and solutions are stored in the classes “Attribute” and “Solution”. It is also possible to define standard values for attributes and solutions and for these values one can store standard distances (as shown in the matrix above). Standard values for an attribute type “material” could be “wood”, “concrete” and “steel” for instance. This property is represented by the class “StandardDistance”. Another class is called “IsPrototypeOf”. This class defines if a case has a prototype (a case which represents a group of cases) and vice versa if a case is a prototype of other cases.

D. Indexing

Experiments with data from “real-world” pointed out an important issue, namely run-time performance. As a case (measurement incl. background information) normally consists of numerous attributes and many distance calculations are necessary to retrieve a set of the most similar cases, it turned out to be necessary to improve runtime performance. Beside multidimensional indices (the current implementation uses an M-Tree), algorithms could be considered to reduce the dimensions of the data-vector and to speed up the system.

Effective and adequate indexing and prototyping can be efficient ways to reduce the runtime of searching similar cases. Groups of cases with very low distances can be represented by prototypes. Due to the fact that comparisons of attributes in this system are metric, the Metric Tree (M-Tree) is a possibility to improve runtime performance by approximately 67% [7]. The algorithm for inserting elements keeps the M-Tree balanced, it grows bottom-up. Figure 10 shows the structure of the Metric Tree and one can see the division of the metric space achieved by the M-tree in Figure 11.

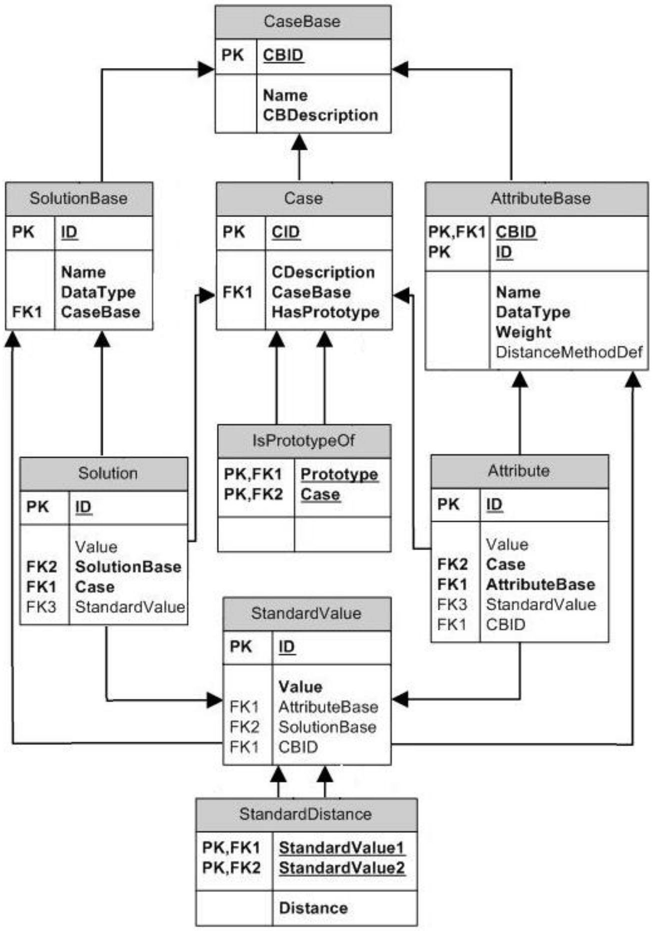


Figure 9. Data Model

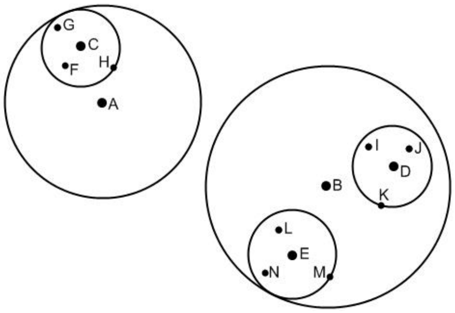


Figure 10. Example M-Tree

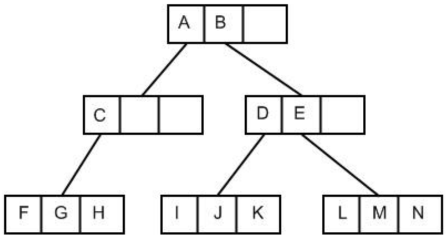


Figure 11. Example M-Tree

The M-Tree divides the space into hierarchically organised clusters. A cluster is represented by the centre and by the so-called partial tree covering radius. In the example above there are two clusters on the highest level (centres A, B) including lower-level clusters (C, D, E) and leaf-nodes (F...N). Relying on this scheme the model can speed up similarity search operations. In case of range queries or k-nearest neighbour queries, the search algorithm of the M-Tree only explores partial trees containing potential candidates and does not consider partial trees (incl. contained objects), where, according to the distance between query object and cluster-centre, also taking the partial tree covering radius into account, a valid search result is not possible.

For more information about the M-Tree model (e.g., insertion, nearest neighbour search) see [7].

VII. CONCLUSION AND FUTURE WORK

Bridge Monitoring is a very complex task. Any building has its individual dynamic parameters which make the automation of measurement analysis and interpretation become quite challenging problems. Aspects like the personal impression of the analyst have influence on the interpretation of measurement results, which by now rarely is represented in a formal way. As mentioned, the Case-based Decision Support System tries to provide decision support to the engineer by pointing out comparable historical measurements. The interpretation of measurement results can be supported well because similar bridges in similar condition have comparable measurement results. Due to using the M-tree model for indexing, the probably big number of entries in the case base, similar cases generally can be provided in a more adequate runtime, although the similarity search can still be a very time-consuming operation. The system described in this contribution currently is in the state of a "research prototype" and mainly provides the functionality to retrieve the most similar structures/objects of a query object.

The main reason for the system requirements explained in this contribution is not a possible redundancy of the expert in the analysing process of Bridge Monitoring but major assistance in his/her work in order to handle the overwhelming amount of measurement data. Besides, these routines might help to attract notice to new aspects which are, due to a lack of time and knowledge, ignored and unknown so far. All in all, the new procedures would support an expert in understanding and classifying the measurement of a bridge and, finally, lead to a better utilization of the measurement data. A design of such system was introduced in this contribution including suggestions for similarity measures and indexing methods.

The outcome of the current investigation is a prototypic implementation of the proposed Case-based Decision Support System for Structural Health Monitoring, drawing conclusions from measurement results (eigenfrequencies) and visual evaluation of buildings to their condition. First experiments with measurements from different types of structures indicated that this is a very promising field of research. Assessing simple structures perform well but the more complex the buildings become (e.g., bridges), the more obviously it turned out that many improvements on the part of computer scientists' methods as well as on the part of civil engineers' procedures are necessary. It may be not enough that the reasoning algorithm just relies on past cases, further rules and constraints might be essential. As an example the fact can be mentioned, that measurement results strongly depend on environmental influences (e.g., weather conditions like temperature, humidity, etc.) for instance, which has to be taken into account in order to be able to draw conclusions from the signal to the building's state more precisely. On the other hand civil engineers would have to improve their inspection procedures in order to collect all data which influences a measured signal and which consequently is important for a computer system to assess a measurement correctly. Nevertheless, the current implementation already can provide support for interpreting signals and in case of evaluating more or less simply designed structures (first experiments with lamp posts were carried out), whereby the output of the Case-based Decision Support Prototype is very close to the engineer's output.

REFERENCES

[1] B. Freudenthaler, G. Gutenbrunner, R. Stumptner, and J. Küng, "Case-based Decision Support for Bridge Monitoring", Proceedings of the Third International Multi-Conference on Computing in the Global Information Technology 2008 – ICCGI 2008, Athens, July 2008.

[2] A. Aamodt and E. Plaza, "Case-Based Reasoning. Foundational Issues, Methodological Variation and System Approaches", AI Communications, vol. 7, no. 1, Amsterdam: IOS Press, 1994, pp. 39-59.

[3] A.A. Angehrn and S. Dutta, "Case-based decision support", Communications of the ACM, New York: ACM Press, 1998, pp. 157-165.

[4] C. Beierle and G. Kern-Isberner, Methoden wissensbasierter Systeme. Grundlagen, Algorithmen, Anwendungen, 2. Auflage, Wiesbaden: Vieweg, 2003.

[5] R. Cantieni, Untersuchung des Schwingungs-verhaltens großer Bauwerke, Technische Akademie Esslingen, 1996.

[6] R.H. Chi and M.Y. Kiang, "An integrated approach of rule-based and case-based reasoning for decision support", Proceedings of the 19th annual conference on Computer Science, New York: ACM Press, 1991, pp. 255-267.

[7] P. Ciaccia, M. Patella, and P. Zezula, "M-tree: An Efficient Access Method for Similarity Search in Metric Spaces", Proceedings of 23rd VLDB Conference, Athens, 1997, pp. 426-435.

[8] S.W. Doebling, C.R. Farrar, M.B. Prime, and D.W. Shevitz, Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in their Vibration Characteristics: a Literature Review, Research Report LA-13070-MS, ESA-EA Los Alamos National Laboratory, Los Alamos, NM, 1996.

[9] J. Eibl, O. Henseleit, and F.H. Schlüter, Baudynamik. Betonkalender 1988, Band II, Berlin: Ernst & Sohn, 1988, pp. 665–774.

[10] C.R. Farrar, K. Worden, G. Manson, and G. Park, “Fundamental Axioms of Structural Health Monitoring”, Proceedings of 5th International Workshop on Structural Health Monitoring, Stanford, CA., September 2005.

[11] B. Freudenthaler, Case-based Reasoning (CBR). Grundlagen und ausgewählte Anwendungsgebiete des fallbasierten Schließens, Saarbrücken: VDM Verlag Dr. Müller, 2008.

[12] B. Freudenthaler, R. Stumptner, E. Forstner, and J. Küng, “Case-based Reasoning for Structural Health Monitoring”, Proceedings of the Fourth European Workshop on Structural Health Monitoring 2008, Cracow, July 2008.

[13] P. Furtner and E. Forstner, BRIMOS Method Statement: Dynamic System Identification and Damage Detection in Bridge Structures, VCE, 2007.

[14] G. Guariso and H. Werthner, Environmental Decision Support Systems, Ellis Horwood series in computers and their applications, 1989.

[15] M.R. Klein and L.B. Methlie, Knowledge-based Decision Support Systems. With Applications in Business, New York: WILEY, 1995.

[16] J.H. Moore and M.G. Chang, “Design of Decision Support Systems”, ACM Press, vol. 12, Issue 1-2, 1980.

[17] R. Myers, R.C. Wilson, and E.R. Hancock, “Bayesian Graph Edit Distance”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 6, pp. 628-635, 2000.

[18] A. Robles-Kelly and E.R. Hancock, “Edit Distance From Graph Spectra”, Proceedings of the Ninth IEEE International Conference on Computer Vision, 2003.

[19] A. Rytter, Vibration Based Inspection of Civil Engineering Structures, PhD thesis, Department of Building Technology and Structural Engineering, Aalborg University, Denmark, 1993.

[20] H.A. Simon, The New Science of Management Decisions, New York: Harper & Row, 1960.

[21] H. Wenzel, Health Monitoring of Bridges, Chichester: John Wiley & Sons Ltd., 2008.

[22] H. Wenzel and D. Pichler, Ambient Vibration Monitoring, Chichester: John Wiley & Sons Ltd., 2005.

ESLAS – a robust layered learning framework

Willi Richert Riccardo Tornese

Abstract— **With increasing capabilities today robots get more and more complex to program. Not only the low-level skills and different strategies for different subgoals have to be specified, which by itself is not a trivial task even for simple domains. Both, the skill set and the strategies, also have to be compatible with each other. This turns out to be a major hassle as they are designed and implemented under assumptions about the future environment and conditions the robot will be faced with, that usually do not hold in reality.**

The *Evolving Societies of Learning Autonomous Systems* architecture (ESLAS) is targeted to this problem. With minimal need for specification, it is able to learn skills and strategies independently in order to accomplish different goals, which the designer can specify by means of an intuitive motivation system. In addition, it is able to handle system and environmental changes by learning autonomously at the different levels of abstraction. It is achieving this in continuous and noisy environments by 1) an active strategy-learning module that uses reinforcement learning and 2) a dynamically adapting skill module that proactively explores the robot's own action capabilities and thereby provides actions to the strategy module. We demonstrate the feasibility of simultaneously learning low-level skills and high-level strategies in a Capture-The-Flag scenario. Thereby, the robot drastically increases its overall autonomy.

Index Terms— **autonomous framework, strategy learning, skill learning, robotics**

I. Introduction

Whenever a robot has to be programmed, its designer has to make many assumptions about the future environment to keep the task tractable. Even more so, if the behavior that the robot will have to exhibit is so complex that it needs different levels of abstractions. The assumptions typically decrease the robot's autonomy and robustness in later application. A learning robot architecture is therefore desirable that places a minimum of assumptions into its algorithms in order to increase its robustness and autonomy. This architecture should combine top-down goal specification with bottom-up exploration of its own capabilities. The desired characteristics of such and architecture are the following:

- The ability to learn and apply continuous actions (skills) in noisy domains.
  - The skill learner should find out by itself what types of capabilities actually are learnable before it starts trying to learn specific skills.
  - Skills should be able to be adapted while being executed.
  - The ability to find a "good enough" action to be executed "fast enough".
  - Skills should provide enough data, methods, and means to categorize observed performance of other robots in order to learn from them for increased learning speed.
  - Skills should abstracted from the strategy and be visible to it only by some kind of *handle*.
- The capability to find state abstractions that are able to distinguish between sufficiently distinct states from the view of the learned skill set while maintaining good generalization.
  - It should account for continuous time.
  - It should support multiple possibly contradicting goals.
  - It should be able to learn from delayed feedback from the environment.

This naturally leads to an architecture consisting of three layers: the motivation, the strategy, and the skill layer. The overall goal can be specified intuitively by different drives that make up the robot's motivation layer. Each drive is representing one sub-goal. The strategy layer has the task to group the infinitely large state space into a small number of abstract regions in order to escape the curse of dimensionality and determine the optimal action for each one of those sub-goals. As the environment can change during runtime, the strategy layer also has to maintain a model about its behavior in that environment. The low-level skills that make up the overall behavior is the task of the skill layer. It has to find out, which actions the robot is actually capable of. It is in charge not only of exploring its own capabilities, but also to optimize them while normally executing them.

Our approach is thus providing a framework that combines strategy learning with Developmental Robotics principles to satisfy the different sub-goals of the overall motivation. In this article, we present a significantly extended version of our previous work [1]. The strategy layer is more autonomous and robust to environmental change in that it does not rely anymore on standard model-based Reinforcement Learning. Instead, it deploys intertwined state abstraction with model-based Reinforcement Learning. The skills are not anymore restricted to fixed models. Instead we fully revised our skill system that allows now for arbitrary models and includes developmental robotics principles in that it is able to learn what is actually able to learn [2], [3].

This work is partially supported by the DFG-Schwerpunktprogramm 1183: Organic Computing

Willi Richert is with Faculty of of Computer Science, Electrical Engineering and Mathematics, University of Paderborn, Germany, Fürstenallee 11, 33102 Paderborn, Germany richert@c-lab.de, Riccardo Tornese is graduate student of Information Technology Engineering of Politecnico di Milano, piazza Leonardo Da Vinci 32, 20133 Milan, Italy riccardo.tornese@mail.polimi.it

II. Related work

When ignoring the need for action recognition, which is necessary for perception-based imitation and coordination, there seems to be a lot of research done in the area of continuous state and action spaces.

A. Model-free approaches

Hasselt and Wiering devised the *Continuous Actor Critic Learning Automaton* approach, which empowers reinforcement learning to operate on continuous state and action spaces [4]. They calculate real valued actions by interpolating the available discrete actions based on their utility values. Therefore, the performance is highly dependent on initial assumptions about the value function.

It is obvious that a full search in continuous state and action spaces is infeasible. For reinforcement learning approaches to be applied in realistic domains, it is therefore vital to limit the search to small areas in the search space. One approach to do that is the *Actor-Critic* method [5]. It separates the presentation of the policy from the value function. The actor maintains for each state a probability distribution over the action space. The critic is responsible for providing they reward from the actions taken by the actor, which in turns modifies its policy. As this relieves the designer from assumptions about the value function, it introduces new assumptions about the underlying probability distribution. To overcome this problem Lazaric et al. devised *Sequential Monte Carlo Learning* [6], which combines the actor critic method with a nonparametric representation of the actions. After initially being drawn from a prior distribution, they are resampled dependent on the utility values learned by the critic.

Bonarini et al. developed *Learning Entities Adaptive Partitioning* (LEAP) [7], a model-free learning algorithm that uses overlapping partitions, which are dynamically modified to learn near-optimal policies with a small number of parameters. Whenever it detects incoherence between the current action values and the actual rewards from the environment it modifies those partitions. In addition, it is able to prune over-refined partitions. Thereby it creates a multi-resolution state representation specialized only where it is actually needed. The action space is not considered by this approach. In their grid world experiment, they use a fixed set of predefined actions.

B. Model-based approaches

The *Adaptive Modelling and Planning System* (AMPS) by Kochenderfer [8] maintains an adaptive representation of both the state and the action space. In his approach, the abstraction of the state and action space is combined with policy learning in a smart way: states are grouped into abstract regions, which have the common property that perception-action-traces, previously performed in that region, “feel” similar in terms of failure rates, duration, and expected reward. It does so by splitting and merging abstract states at runtime. AMPS not only dynamically abstracts the state space into regions, but also the action space into action regions. This is, however, done in a very artificial way that could not yet been shown to work in real world domains.

Although our strategy layer is inspired by AMPS, we differ from it in the following important points: AMPS applies the splitting and merging also to the action space, which works fine in artificial domains but will not cope with the domain dependency one is typically faced with in real environments. In contrast to that, we use goal functions as the strategy’s actions, which have to be realized by a separate skill-learning layer. This leads to a perfect separation of concerns: the task of the strategy layer is to find sequences of actions and treats actions as mere symbols. The skill layer by means of data driven skill functions then grounds these symbols.

Another aspect is the supported number of goals. Take for example a system, which has to fulfill a specific task while paying attention to its diminishing resources. While, on the one hand, accomplishing the task, the resources might get exhausted. If it, on the other hand, always stays near the fuel station, the task will not be accomplished. Approaches like AMPS, which do not support multiple goals by multiple separate strategies, have to incorporate all different goal aspects in one reward function. This leads to a combinatorial explosion in the state space and implicates a much slower learning convergence.

As already described, we use abstract motivations, which the designer has to specify. These motivations may also contain competing goals. The major advantage of our approach is that the robot can learn one separate strategy for each motivation. Depending on the strength of each motivation, it has now a means to choose the right strategy for the actual perception and motivation state.

C. Discussion

All these approaches have the following underlying restricting assumptions. First, they assume that optimal actions are either possible to be predefined or effectively learnable within the reinforcement learning framework. That means that prior to using these approaches a careful analysis of all occurring events in the environment has to be carried out by the designer. Except for AMPS, they are all based on Markov Decision Processes (MDP). Time varying actions, which are the norm in realistic scenarios, however require a semi-Markov Decision Process (SMDP), which complicates the search in continuous action spaces. Arguing that models are difficult to approximate at runtime the model-free approaches do not learn a model on which the policy is approximated but only the value function. Furthermore, they always solve only one goal and it is not intuitively clear, how multiple possibly contradicting goals could be integrated using the same state and action space for all goals. The biggest problem of all, however, is that these approaches are solely aimed at learning from scratch. It is not clear how those could be combined with imitation or coordination – aspects, which are vital to application multi-robot scenarios. The ESLAS architecture, which will be described in the following, was designed with these aspects in mind.

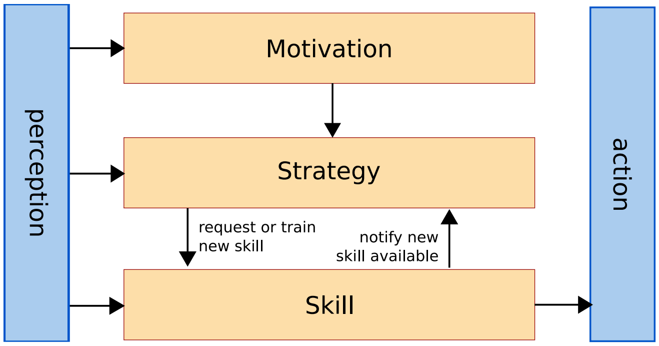


Fig. 1. The architecture of ESLAS consists of three layers. Every layer can read the perception and send its output to the module below itself

III. The ESLAS architecture

Vital to the multi-robot imitation approach is the *Evolving Societies of Learning Autonomous Systems* (ESLAS) architecture that supports it by means of its layered recognition approach [9]. Thereby, we extend our previous layered architecture [1] to be usable for imitation in multi-robot scenarios.

The ESLAS architecture is based on three layers of abstraction as shown in Fig. 1. At the top level, a motivation layer provides a motivation function for the learning algorithm being the overall goal of the robot. This function determines which goal is the most profitable one to reach at each moment. With different motivations, the learning algorithm is able to handle changes in the environment without the need of relearning everything. At the medium layer is the Reinforcement Learning algorithm, which incorporates the method from AMPS of state space revising in parallel to SMDP policy calculation. It receives input from the interface and decides which skill is executed. A skill is described by a goal function and handled in the lowest layer. Skills can be simple, like driving forward, but also quite complex, depending on this function. Using this function, a skill is also capable of recognizing whether a skill similar to itself has been executed in the observations.

A. Motivation layer

For the evaluation of the robot's overall state, we use biologically inspired evaluation methods similar to emotions. With that, we specify all high-level goals in the form of a motivation system (Fig. 2):

$$\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)^T \ , \quad \mu_i \in \mathbb{R}^+ \ . \qquad (1)$$

Each motivation μ_i corresponds to one high-level goal, which is considered accomplished or satisfied if $\mu_i < \mu_i^\theta$, with μ_i^θ defining the threshold of the *well-being region* (Fig. 3). By specifying $\mu_i : \mathbb{S} \rightarrow \mathbb{R}$ as a mapping from the strategy's state space to the degree of accomplishment of goal i and μ_i^θ as the satisfaction-threshold of that goal the designer is able to intuitively define the robot's overall goal, which it accomplishes by minimizing each motivation's value. When it is adapting its strategy or skill set, it does so with only this urge in mind.

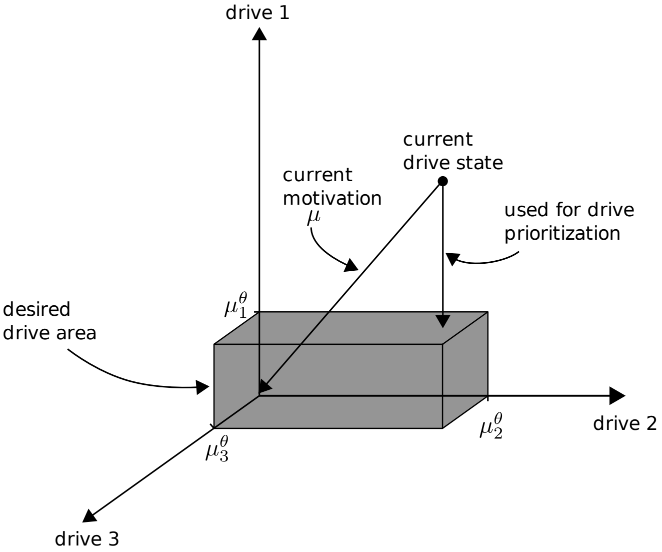


Fig. 2. The motivation system: each drive measures the status of accomplishing one sub-goal with zero being fully accomplished. The current motivation is the vector to the point of origin

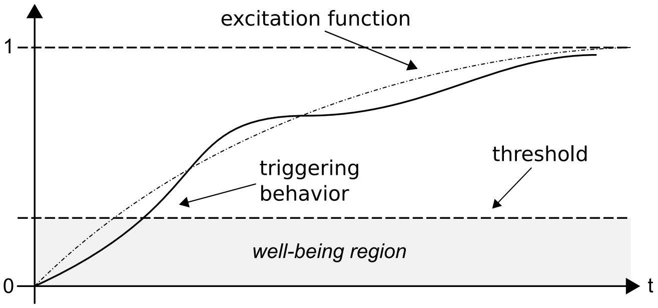


Fig. 3. An example of specifying a sub-goal by means of the motivation system's drive. The excitation function describes the force the current drive state is subject to. By specifying it dependent on the perception and on the internal state of the robot the user is "programming" the final behavior

If the vector of the current drive state to the point of origin is interpreted as the current motivation, it serves two functions in the ESLAS framework: on the one hand $-\dot{\boldsymbol{\mu}}$ is used as a reward for the strategy layer, which will be described in the next section. On the other hand, it supports imitation in multi-robot scenarios: the motivation value in this motivation layer can be used to express the robot's overall well-being to the other robots and guides them when they are observing each other to imitate only obviously beneficial behavior.

B. Strategy layer

In order to satisfy the motivation layer the robot has to learn a strategy that is able to keep $\boldsymbol{\mu} < \boldsymbol{\mu}^{\boldsymbol{\theta}}$, given only the experience stream

$$\ldots, (o, a, d, \boldsymbol{\mu}, f)_{t-1}, (o, a, d, \boldsymbol{\mu}, f)_t, \ldots \qquad (2)$$

where o_t is the raw observed state, a_t the executed action triggered in the last time step, d_t the duration of that action, f_t signals whether the action has failed, which will be described later on. To keep this learning program tractable the strategy layer is not trying to learn one strategy for the whole motivation system. Instead, it is generating one strategy for each motivation. The system then selects the strategy to follow dependent on the dynamic drive prioritization $\max(\mathbf{0}, \boldsymbol{\mu} - \boldsymbol{\mu}^{\theta})$. In the current approach it chooses the drive with the least satisfied motivation.[1]

In the following, we will restrict the description to one strategy. It will have to generalize the actual state observations into abstract regions on which it then uses Reinforcement Learning to find a sufficiently good strategy, as operating on the raw state space would be unfeasible. Thereby, it can be used with any form of abstraction method. In this work, we use nearest neighbor [10]. The environmental model is updated during runtime as new experience is made by the robot and thus subject to change. A model-based Reinforcement Learning with prioritized sweeping [11] is used to derive an optimal policy by means of semi-Markov decision processes (SMDP) [12], [13]. Model-free Reinforcement Learning methods like Q-learning [14] are not practicable in this case, because all experience gets lost each time the underlying model changes.

We will now present the three main components of the strategy layer and their interaction (Fig. 4): the processed, filtered and purified perception is stored as an interaction sequence, which we will call *experience*. It is modified by several heuristics to build a *model* upon which the *policy* is generated. We will start explaining how the strategy is learned, if the model has already been built so that it reflects the experiences and abstracts the raw state observations o to abstract state regions $s \in \mathbb{S}$.

1) Policy: A policy is a mapping

$$\pi : \mathbb{S} \rightarrow \mathbb{A} \tag{3}$$

that assigns each state $s \in \mathbb{S}$ an action $a \in \mathbb{A}$. Let $V^{\pi} : \mathbb{S} \rightarrow \mathbb{R}$ be a value function that estimates "how beneficial" it is for a robot to be in a given state. A policy π is called optimal, if $V^{\pi}(s) \geq V^{\pi'}(s) \; \forall \; s \in \mathbb{S}$. Thereby, it maximizes the expected long-term discounted sum of rewards [13] and is denoted by π^*. The reward is discounted, as it is wise to give a smaller weight to reward that is further away in the future.

As in real-world scenarios the duration of actions are variable, the discounting is done continuously by $\beta \in (0, \infty)$: a reward r received after time t thus leads to a net reward of $e^{-\beta t}$. If $\beta = \infty$ the robot is said to be myopic, as the future reward is discounted by $e^{-\infty t} \approx 0$ and the robot thus is concentrating only on the immediate reward. With β approaching zero the robot is paying more and more attention to reward that is farther in the future.

The reward in our work is composed of two reward elements. The lump sum reward r specifies the one time reward for transferring the robot from the abstract state s with action a to the abstract state s'. The reward rate ρ is given continuously for staying in state s while executing action a until the robot arrives at state s. This is necessary to provide the most general form of goal specification via the motivation system. Both components can be extracted from the motivation by means of

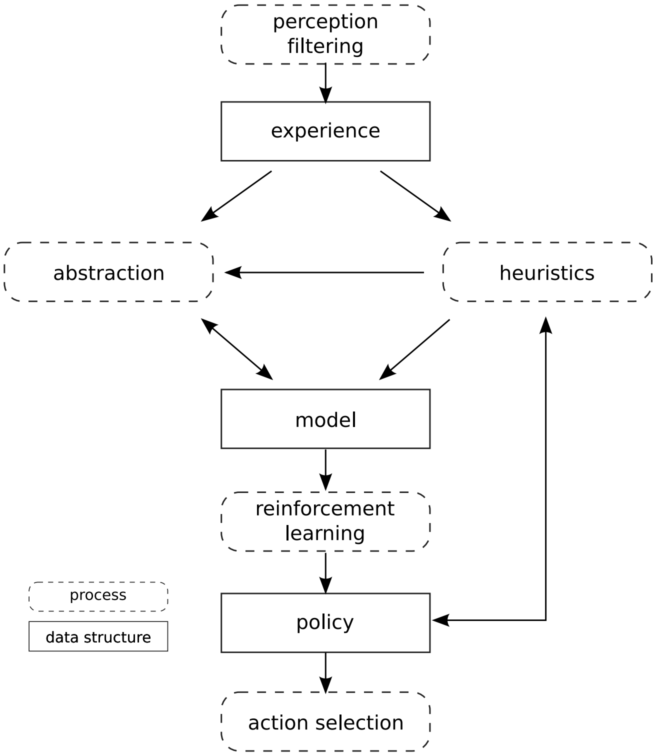


Fig. 4. Processes involved in the strategy layer

$$r, \rho = \begin{cases} (-\dot{\mu}_i, 0) & \text{if } |\dot{\mu}_i| > \rho_\theta \\ (0, -\dot{\mu}_i) & \text{otherwise} \end{cases} . \tag{4}$$

That means that the reward is interpreted as lump-sum reward, if it exceeds the reward rate threshold ρ_θ, otherwise it is received as reward rate.

In the following, we will stick to the notation of Kochenderfer regarding the learning of strategies on abstract state spaces with SMDPs. With $P_t(t \mid s, a, s')$ being the probability that it takes at most t time to move from s to s' by means of executing action a the *discounted value of the unit lump-sum reward* is calculated as[2]

$$\gamma(s, a, s') = \int_0^{\infty} e^{-\beta t} dP_t(t \mid s, a, s') \; . \tag{5}$$

The *average cumulative discounted sum of the reward* received continuously while executing action a in state s until arriving at state s' is calculated as

$$\lambda(s, a, s') = \int_0^{\infty} \int_0^{t'} e^{-\beta t'} dt' dP_t(t \mid s, a, s') \; . \tag{6}$$

[1] In addition, we are investigating methods, which try to detect situations in which compromises are made by choosing a possibly suboptimal action for one motivation, if others can be pleased with that action as well.

[2] Not to be confused with the discount factor γ in MDP problems.

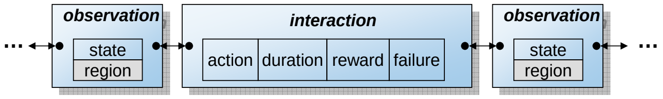


Fig. 5. The interaction sequence forms the experience flow

The expected discounted reward when started in state

$$V_\pi(s) \equiv E\Bigg\{ \sum_{k=1}^{\infty} \left[e^{-\beta t_{k+1}} r_k + \int_{t_k}^{t_{k+1}} e^{-\beta t} \rho_k dt \right] \Big| \quad (7)$$

$$s_1 = s, a_k = \pi(s_k) \Bigg\}. \quad (8)$$

To find the optimal value function $V^*(s)$, the robot is updating recurrently the value function each time a new event occurs using

$$V(s) \leftarrow \max_{a \in \mathbb{A}} \left[R(s,a) + \sum_{s' \in \mathbb{S}} P(s'|s,a) \gamma(s,a,s') V(s') \right]. \quad (9)$$

Eventually, $V(s)$ will then converge to the true value function $V^*(s)$ [8]. The optimal policy can then be computed similarly:

$$\pi^*(s) = \arg\max_{a \in \mathbb{A}} R(s,a) + \sum_{s' \in \mathbb{S}} P(s'|s,a) \gamma(s,a,s') V^*(s') \quad (10)$$

The only thing left to do is to estimate $R(s,a)$ ("(s,a,s')" omitted):

$$R(s,a) = \sum_{s' \in \mathbb{S}} P(s' \mid s,a) \left(\gamma r + \lambda \rho \right) \quad (11)$$

Kochenderfer showed that this can be done with non-parametric estimation [8]. For that it is first necessary to estimate $\gamma(s,a,s')$, as $\lambda(s,a,s')$ simplifies to $\lambda(s,a,s') = (1 - \gamma(s,a,s'))/\beta$. If $n(s_k, a_k, s_{k+1})$ is the number of (s_k, a_k, s_{k+1}) transitions, then γ is estimated after the k^{th} transition as follows:

$$\hat{\gamma}(s_k, a_k, s_{k+1}) \leftarrow \hat{\gamma}(s_k, a_k, s_{k+1}) + \frac{e^{-\beta t_k} - \hat{\gamma}(s_k, a_k, s_{k+1})}{n(s_k, a_k, s_{k+1})} \quad (12)$$

If $\sigma_r(s,a,s')$ is the accumulated lump-sum reward and $\sigma_\rho(s,a,s')$ the sum of the reward rates received when going from s to s' with action a, then the estimated expected reward for executing a in s can be calculated as

$$\hat{R}(s,a) = \frac{1}{n(s,a)} \sum_{s' \in \mathbb{S}} \left(\hat{\gamma}(\sigma_r - \sigma_\rho/\beta) + \sigma_\rho/\beta \right) \quad (13)$$

2) Experience: To arrive at a tractable number of meaningful states, the raw states have to be abstracted first. The ESLAS approach works on interactions that describe the action in and the reaction of the environment (Fig. 5). An interaction encodes the following data:

- *action:* This is the output that was delivered to the action tower in the last step.
- *duration:* The strategy is not informed every time new information is available. Instead, it is triggered using intelligent heuristics, which will be described later on.
- *reward:* The return of the last action in the form of the motivation vector.
- *failure:* This is retrieved from the skill layer and denotes whether the skill executed in the last step estimates the outcome as success or failure. It will be described in more detail in Sec. III-C.

An interaction is always connected to its starting and ending states (Fig. 5) provided by the perception. All the experience is saved in an experience list, which consists of *interactions*. We define an interaction to describe the important data of one time frame

$$I_{t_1}^{t_2} = (o_{t_1}, a_{t_1}, d_{t_1}, r_{t_1}, f_{t_1}, o_{t_2}), \quad (14)$$

where o_{t_1} and o_{t_2} are the raw state observations at the beginning and end of a time frame (*state* in Fig. 5). a_{t_1} is the executed action, r_{t_1} the reward vector received by the motivation layer and $d_{t_1} = t_2 - t_1$ the duration. Finally, f_{t_1} denotes a failure of the last step. This can be e.g. the skill layer signalling that the previously executed skill has not performed as expected, because the robot is trying to drive against a wall. For realistic applications, it must be taken care that the robot is not spammed with uninteresting information. A new interaction is generated if one of the following heuristics holds:

- The perception signals a sufficiently different state by some distance metric: $d(o_{t_1}, o_{t_2}) > \theta_o$.
- The motivation layer has signaled a sufficiently interesting motivation change: $r_{t_1} = |m_{t_2} - m_{t_1}| > \theta_r$
- A certain amount of time has passed: $t_2 - t_1 > \theta_t$

3) Model: At the beginning, all states belong to only one region, as the robot has no reason to believe otherwise. While interacting with the environment the model is modified by several heuristics, which are invoked recurrently to maintain a mapping of observations in the perception space $\mathbb{R}^d$ (*state* in Fig. 5) to states in the abstracted region space $\mathbb{S}$ (*region* in Fig. 5). d is the number of dimensions of the perception space[3]. The heuristics split or merge regions so that the model and underlying statistics reflect the world experience. The following heuristics are found to be necessary.

a) Transition heuristic: As mentioned above, the continuous state space is split into regions so that for each raw state belonging to the same region executing the same action "feels" similar to the robot. That requires that $Q(s,a,s')$ as the value for transitioning from s to s' with the greedy action $a = \pi(s)$ can be estimated with a sufficient confidence. This is calculated using interaction sequences starting in s and arriving in s' while only executing the greedy action a:

$$\begin{aligned} Q(s,a,s') &= \gamma(s,a,s')(r(s,a,s') + V(s')) \quad (15) \\ &\quad + \lambda(s,a,s')\rho(s,a,s') \quad (16) \end{aligned}$$

Let $succ_a(s) = \{s' \mid P(s' \mid s,a) > 0\}$. If raw states are mistakenly grouped into the same abstract region the variance of the $Q(s,a,s')$ values calculated for all the greedy traces

[3] In the experiments nearest neighbor is used. Practically any abstraction mechanism can be used that supports add/remove/query at runtime.

belonging to the same region will increase. A high variance indicates that splitting that region will likely lead to better transition estimates in the split regions:

$$Var\left(\{\, Q(s,a,s') \mid s' \in succ_a(s) \,\}\right) > \theta_{TV} \qquad (17)$$

This is done by clustering the traces so that traces with similar $Q(s,a,s')$ grouped together. For each cluster, one region is created.

The challenge lies in determining θ_{TV}. AMPS requires the designer to analyze the scenario and empirically determine that value beforehand. This is apparently no possibility for groups of robots, which have to learn the proper behavior autonomously themselves. As the distribution for Q usually cannot be foreseen it happens that θ_{TV} is either to low, which results in too fine state abstraction and slows down the learning speed, or too high which leaves too much aliasing in the strategy. The competing forces for determining θ_{TV} are as follows:

1) The more often the robot is experiencing aliasing and the higher the variance of the resulting regions' values is, the higher the inclination to split should be.
2) The lower the variance is compared to the maximum region value the lower the inclination to split should be.

The first point is solved by using $QVar$ which weights the deviation from the mean by the region's transition probability. The second is handled by normalizing both mean and the Q-values to the maximal region value $V_{max} = \max(\{V(s) \mid s \in \mathbb{S}\})$. With the following definition for $QVar$ the threshold θ_{TV} can be set to a fixed value without having to bother about the future development of the region values:

$$QVar\left(Q(s,a,s')\right) \equiv \sum_{s' \in succ_a(s)} \frac{P(s' \mid s,a)}{V_{max}^2 \cdot |succ_a(s)|} \cdot \left(\overline{Q(s,a,s')} - Q(s,a,s')\right)^2 \qquad (18)$$

The inclination to split is thus adapting with the changing value function at run-time.

b) Experience heuristic: This heuristic limits the memory horizon of the robot to θ_M interactions. It removes interactions that are too far in the past in order to keep the robot's model and policy more aligned to the recent experience of the robot. Basically, it removes those old interactions from its memory and adds the new experience to it. Thus, it is modifying the experience of at most two regions, which might cause an update of the model and of the policy.

c) Failure heuristic: A failure rate is associated with each region. It describes the ratio of failure signals when the greedy action of the corresponding region has been executed to the number of success signals. These signals are emitted by the strategy and skill layer which will be described later on. They are encoded as f_t in the interactions (Eq. (14)). Failure signals are scenario specific and can be emitted if e.g. the robot bumps into a wall or if it has not encountered something interesting for a longer period of time. The failure heuristic splits a region if its greedy action's failure rate is not homogeneous enough:

$$\theta_f < f < 1 - \theta_f, \quad (0 < \theta_f < 1/2) \qquad (19)$$

The lower the user defined threshold θ_f is, the more eager the failure heuristic is trying to split a region. This forces the state abstraction to arrive at regions that have failure rates with which a more deterministic strategy can be computed. For both resulting new regions individual greedy actions can then be determined by the reinforcement learning algorithm.

d) Reward heuristic: Especially in the beginning of the robot's lifetime, when there is not yet enough information for the transition and simplification heuristic to adapt the state space based on sufficient statistical data, the reward heuristic is of importance. It allows a region $s \in \mathbb{S}$ to be split if the reward rate variance is too high. This indicates that the action performed in that region gives a too diverse feedback. A split of that region will then lead to multiple regions, which are more consistent with regard to the expected reward rates. This also is vital in cases where the failure signal is too seldom, as it provides the only other possibility to initially split a region.

In particular, the reward heuristic is looking in the reward rate stream for a clear switch from low to high variance areas, where both areas are of sufficient length. Only such a switch in variance indicates clearly that a split is advisable. Therefore the reward heuristic considers the reward rates of the last n interactions made in the current region. The lump sum rewards in that time frame are not considered, as they will show non-zero values only in rare occasions. Let $\rho_{t_1}^{t_2} = (\rho_{t_1}, \ldots, \rho_{t_2})$ and t be the time at which the split is considered. The reward heuristic is searching for an index k that splits ρ_{t-n}^{t} into the two sequences ρ_{t-n}^{t-k-1} and ρ_{t-k}^{t}, such that the following condition holds:

$$\begin{gathered}\left(Var(\rho_{t-n}^{t-k-1}) \approx 0 \;\wedge\; Var(\rho_{t-k}^{t}) > \theta_{RV} \;\wedge\; |\rho_{t-n}^{t-k-1}| > \theta_l\right) \\ \bigvee \qquad (20) \\ \left(Var(\rho_{t-n}^{t-k-1}) > \theta_{RV} \;\wedge\; Var(\rho_{t-k}^{t}) \approx 0 \;\wedge\; |\rho_{t-k}^{t}| > \theta_l\right)\end{gathered}$$

The minimum variance threshold θ_{RV} is dependent on the motivation system design. Recall from Sec. III-B.1 that the reward received by the motivation system is interpreted as a reward rate, if $|\dot{\mu}_i| \leq \rho_\theta$. With $\theta_{RV} = k \cdot \rho_\theta$, $(0 < k < 1)$, a switch is easily detected by the reward heuristic. The minimum low variance sequence length θ_l ensures that the reward heuristics does not find trivial splits. Naturally it is set to be a fraction of the considered time horizon n.

e) Simplification heuristic: As splitting might lead to overly complex models a means is needed that again merges regions once the robot has gathered new experience that suggests a simpler model. This is the task of the simplification heuristic, which analyzes sequences of regions connected by greedy actions. Similar to AMPS we consider here chain and sibling merges. Let a behave nearly deterministically in s, then $succ(s,a)$ denotes the region the execution of a leads to:

$$succ(s,a) \equiv \begin{cases} s' & \text{if } P(s'|s,a) \approx 1 \\ \text{None} & \text{otherwise} \end{cases} \tag{21}$$

A chain merge of two regions s' and s' is performed if

$$succ(s',\pi(s')) = s'' \wedge succ(s'',\pi(s'')) = s \wedge \pi(s') = \pi(s'') \; . \tag{22}$$

In this case the region s'' is superficial and can thus be merged with s' into the new region $s''' = s' \cup s''$, with $succ(s'''',\pi(s''')) = s$ and $\pi(s''') = \pi(s`) = \pi(s'')$. All other regions that resulted into either s' or s'' are updated accordingly.

In the same vein a sibling merge is triggered if

$$succ(s',\pi(s')) = s \wedge succ(s'',\pi(s'')) = s \wedge \pi(s') = \pi(s'') \; . \tag{23}$$

In this case s' and s'' have similar expectations about the future region if the same action is executed.

So far we have assumed that $\mathbb{A}$ is always provided beforehand and that the strategy simply has to choose the right action at each state. For real-world scenarios it would be advantageous if also $\mathbb{A}$ could be learned at run-time. AMPS does this by applying to $\mathbb{A}$ the same abstraction heuristics that helped to organize the state space $\mathbb{S}$. The actions learned in this way, however, are limited to simple domains, where the real-world dynamics can be presented by simple hypotheses. In the next section the *Automatic Modular Action Framework* (AMAF) is presented, which is able to learn reactive actions that are robust to noise and can handle complex dynamics.

C. Skill layer

The skill layer provides a generalized learning method for learning reactive low-level skills. It offers to the strategic layer two working modalities, one for training new skills and one for executing one of the learnt skills. As long as no skills are available, the skill layer explores the space of the low-level actions by composing with random values the output vector that will be sent to the actuators.

Each learnt skill allows to control the perceived properties of the environment by continuously associating an error value to the input data. The learnt skills are communicated to the strategic layer through the identifier that will be used to handle the skill and the definition of the skill.

When the execution of a skill is requested, the skill layer reacts to the received inputs with low-level actions (output vectors) that minimize the error described in the definition of the skill.

For our strategy-learning algorithm, we assume that all skills have finished building hypotheses and are ready for execution.

There are two major benefits with the skill layer over using atomic actions. First, it gives the possibility to automatically create quite complex actions that evolve and adapt to changes. Second, a skill can recognize itself out of a trace of observations by monitoring the value of the error. If this decreases, it is reasonable to deduce that the action was in execution. This is how we solve the correspondence problem between observed execution of foreign behavior and own capabilities in our scenario [15].

The skill layer has been implemented using AMAF (Automatic Modular Action Framework), a framework for the automatic creation of abstract actions. The generated abstract actions will be used as skills by the strategy. What follows is an overview of the framework.

IV. AMAF Specification

A. Working modalities

AMAF can work in two modalities: the passive (or execution) and the active (or training) modality:

- *Execution:* during the execution modality, AMAF is passive because the robot decides which action to execute.
- *Training:* during the training modality, AMAF is active because it decides which action to execute in order to experiment new actions.

This distinction has not to be confused with the one between learning and acting. Learning and acting are parallel processes indeed: AMAF supports the learning phase during both the execution and training modalities. During both modalities, AMAF has to react to the input data with an output vector, so the acting is always enabled.

B. Environment structure

AMAF works with structured data in order to have more abstract and powerful actions and to make a faster and more general learning possible. The robot's perception has to be structured in *objects* and *properties* in order to be used by AMAF. A property is an attribute of an object recognized in the environment. It is composed by an ID and a value, expressed by a real number. An object is an element of the environment characterized by an ID and a set of properties.

The input data has to be structured as a list of tuples $\langle id_o, id_p, v\rangle$ where id_o is the identifier of an object o in the environment, id_p is the identifier of a property p of o describing one perceived attribute of that object, and $v \in \mathbb{R}$ denotes its value.

C. Output representation

AMAF generates abstract actions starting from the actions of the lowest level of abstraction. Usually in robotic applications, the low-level action is the set of intensities of the electrical signals sent to the actuators or, in the case of servomotors, the sent value. A general way to interpret the information sent to an actuator is the effort that actuator will make. AMAF represents a low-level motor action by the output vector $\boldsymbol{M} = (m_1,\ldots,m_n)$, with $m_i \in \mathbb{R}$, $-100 \leq m_i \leq 100$, indicating the effort that a certain actuator will make, and n being the number of actuators the robot controls.

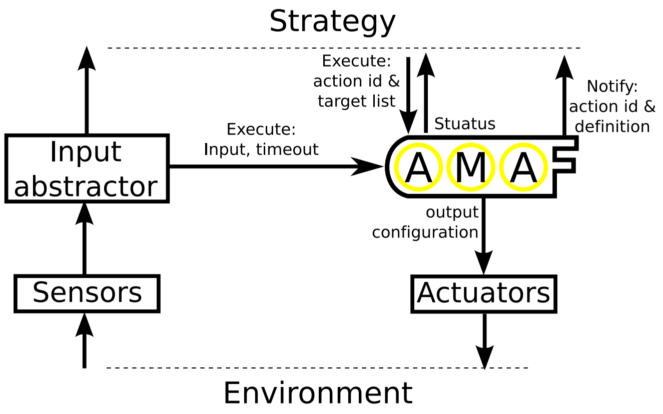


Fig. 6. AMAF interaction with the other components. The row input data is supposed to be elaborated by a specific component called "input abstractor"

D. Interfaces

The strategy communicates with AMAF by sending the action to be executed. There is just one pre-built action with ID *train* that switches AMAF to the active modality. All the other actions are learned by AMAF during its execution. The robot executes a learned action by sending to AMAF the action ID and the list of the targets of the action. These are expressed by the IDs of the objects that the action is applied to. During this phase, the robot can monitor the status of the current action. When the action is set, the status is *starting*. After the first low-level interaction with the environment the status can be *failure* if it is not possible to execute the action, *execution* if the action is executed at low level but the success condition is not verified or *success* if the success condition has already been reached. During both active and passive phases, AMAF can decide that an action executed during the training phase is *mature*. This indicates that it is ready to be immediately used by the robot. In this case, the action is notified to the robot by sending the action ID and the action definition.

At the interface to the environment, AMAF receives the abstract input data and returns the output configuration that realizes the abstract action. It has to compute the solution in a certain time interval that is dependent on the frequency of interaction with the environment. In order to react meaningful in time, AMAF receives in addition to the abstract input the timeout for searching the best low-level action. After the timeout, AMAF has to return the best output configuration found by then, even if it is not globally the best for the last received input.

Fig. 7 gives an example of a low-level interaction. The sensors send the raw input data to an external component, called "Input Abstractor". This processes the raw input and computes information usable for AMAF in a certain time interval. A timeout determines how much time AMAF has for calculating the output configuration. Finally, the output configuration is sent to the actuators. The interaction cycle starts again when a new input is perceived from the sensors and sent to the Input Abstractor.

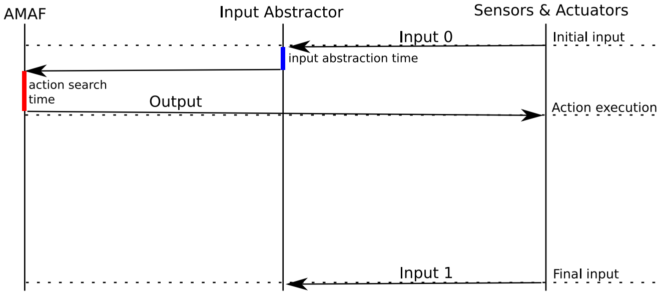


Fig. 7. Example of low level interaction

For each interaction with the environment, AMAF assumes that the next input is the effect of the returned output on the actual input. This can be a good approximation only when the time interval between the initial input perception by the sensors and the execution of the output configuration by the actuators is not relevant in comparison to the time interval between the two input perceptions. We can usually consider unimportant the latencies of the sensors, of the actuators and of the communication between the components. It is sufficient to take into account just the computational time of the Input Abstractor ΔT_I and of AMAF ΔT_M, and the time interval between two inputs ΔT_i. More formally this means that $\Delta T_I + \Delta T_M \ll \Delta T_i$.

E. Actions

The aim of AMAF is to automatically generate abstract actions that allow the robot to move easily in the state space and accomplish its tasks. Each robot, however, can have a different state space representation. The way, in which the input variables are interpreted and combined together, varies significantly from the different implementations of the robot. I.e., AMAF cannot directly control the robot state through its actions. The AMAF solution is to control the input variables. The robot state space is directly built on their values so actions that control the input variables, indirectly allow any kind of robot to move freely in its state space.

Each perceived property of an object is controlled by a "basic action". In addition, AMAF is able of generating "complex actions" that coordinate the execution of different basic actions. The actions generated by AMAF are *multi-target* because it is possible to specify the list of identifiers of the objects that are target of the action. In this way, the complex action can control the value of the properties of different objects at the same time. We will now give a detailed definition of basic and complex actions.

1) Basic Actions: The elementary blocks of the actions supplied by AMAF are the "basic actions". A basic action is defined by a tuple $\langle c, p \rangle$ where p is a property of the object specified as target and c indicates a control. A *control* is a function $f_c : \mathbb{R}^2 \rightarrow \mathbb{R}^+$ that associates an error to each tuple $\langle iv_p, av_p \rangle$ where iv_p is the value of p when the action was started and av_p is the current value of p. The function $f_e : \mathbb{R} \rightarrow \mathbb{R}^+$ obtained by fixing the value iv_p is called *error function* of the basic action. During the execution of the basic action, AMAF tries to decrease as much as possible the value of the error so different controls determine different behaviors. E. g., if f_e is proportional to the value of the property, the basic action decreases its value. On the opposite if f_e is inversely proportional to the value of the property, the basic action increases its value. It is even possible to have actions that control the variation of the value of the property. E. g., in order to specify an action that increases iv_p of an interval Δp it is sufficient to use a control $f_c(iv_p, av_p) = |iv_p + \Delta p - av_p|$.

2) Complex Actions: Basic actions allow the robot to move in the state space but sometimes there can be performance requirements that cannot be satisfied by simply executing basic actions in sequence. It can be for instance necessary to execute two basic actions at the same time or to start executing one action when another one is going to finish. Imagine a task that requires to get close to an object and to shoot it. In this case it is necessary at first to reduce the distance to the object then it is necessary to center the object and finally to shoot. The result can not be efficient if the actions are executed in sequence, but if the object is reached while centering it, the chances of success increase. In AMAF it is possible to coordinate different basic actions through the “complex actions”.

The number of targets and the ordered sequence of steps define a complex action. Each step is defined by a tuple $\langle l_R, t_R, s_R, l_A\rangle$ where l_R is a list of references that determine the *reference error*, $t_R \in \mathbb{R}^+$ is the *success condition* threshold, $s_R \in \mathbb{R}^+$ is the *precondition* threshold and l_A is the list of the weighted basic actions. The elements of the list l_R are tuples $\langle c, p, t, w\rangle$ where c is a control, $t \in \mathbb{N}^+$ indicates the index of the object o of the target list, p is a property of o and $w \in \mathbb{R}^+$ is the weight of the reference. The reference error is computed by linear combination of the errors obtained by applying the controls on the input data. The weights are used as coefficients of the linear combination. The value of reference error V_r determines the value V_{pl} of the progress level:

$$V_{pl} = \begin{cases} 0 & \text{if } V_r > s_R \\ \frac{s_R - V_r}{s_R - t_R} & \text{if } t_R < V_r < s_R \\ 1 & \text{if } V_r < t_R \end{cases} \qquad (24)$$

When the value of the progress level is one, the success condition of the step is reached and the execution of the step can be considered completed.

The elements of l_A are weighted basic actions a defined by the tuple $\langle c_a, p_a, t_a, cf_a\rangle$ where c_a is a control, $t_a \in \mathbb{N}^+$ indicates the index of the object o of the target list, p_a is a property of o and $cf_a : [0,1] \rightarrow [0,1]$ is the coefficient function that determines the value of the coefficient for each value of the progress level. The coefficient function $cf_a(V_{pl}) = start + f_s(V_{pl}) \cdot scale$ is defined by the tuple $\langle id_s, scale, start\rangle$ where id_s is the identifier of the shape function $f_s : [0,1] \rightarrow \mathbb{R}$, $scale \in \mathbb{R}$ is the scaling factor and $start \in \mathbb{R}$ is the starting value of the coefficient. The value of the coefficient c_a of the basic action a is determined by coefficient function and the progress level:

$$c_a(V_{pl}) = \begin{cases} 0 & \text{if } cf_a(V_{pl}) < 0 \\ cf_a(V_{pl}) & \text{if } 0 \leq cf_a(V_{pl}) \leq 1 \\ 1 & \text{if } cf_a(V_{pl}) > 1 \end{cases} \qquad (25)$$

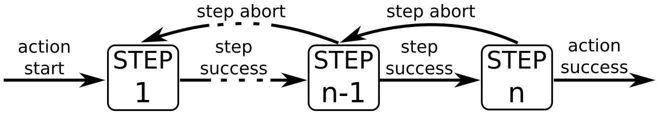


Fig. 8. A complex action is composed by a sequence of steps

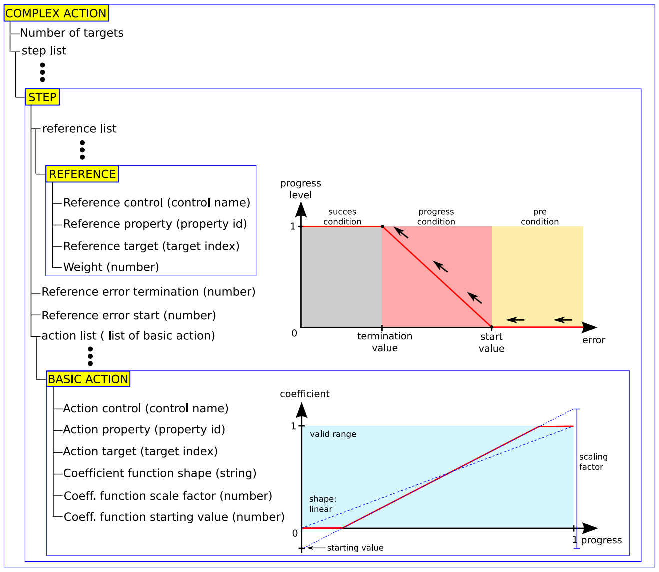

COMPLEX ACTION
Number of targets
step list
STEP
reference list
REFERENCE
Reference control (control name)
Reference property (property id)
Reference target (target index)
Weight (number)
Reference error termination (number)
Reference error start (number)
action list (list of basic action)
progress level
1
0
succes condition
progress condition
pre condition
termination value
start value
error
BASIC ACTION
Action control (control name)
Action property (property id)
Action target (target index)
Coefficient function shape (string)
Coeff. function scale factor (number)
Coeff. function starting value (number)
coefficient
1
0
valid range
shape: linear
scaling factor
starting value
1
progress

Fig. 9. The structure of the definition of a complex action

Calling A the set of basic actions a executed during the step, the error function of the step is

$$f_s = \sum_{a \in A} c_a(V_{pl}) e_a(p_a) \;, \qquad (26)$$

where e_a is the error function of a.

The action starts with the execution of the first step. When the success condition of the step is reached, the second step is executed and so on until the last step. The success condition of the complex action is reached when the last step is in its success condition as shown in Fig. 8. The structure of the definition of a complex action is shown in Fig. 9.

F. Framework structure

The modules of AMAF are divided into two groups: the learning modules and the performing ones. The former have to learn from the interaction with the environment and with the robot. Each learning module represents its knowledge through specific elements called *knowledge units* and gives a score to each of them. The performing modules for the execution of both passive and active modalities use these elements.

The performing modules directly communicate through buffers while the information flow of the learning process is based on the CENTRAL LOG SYSTEM . This is a component

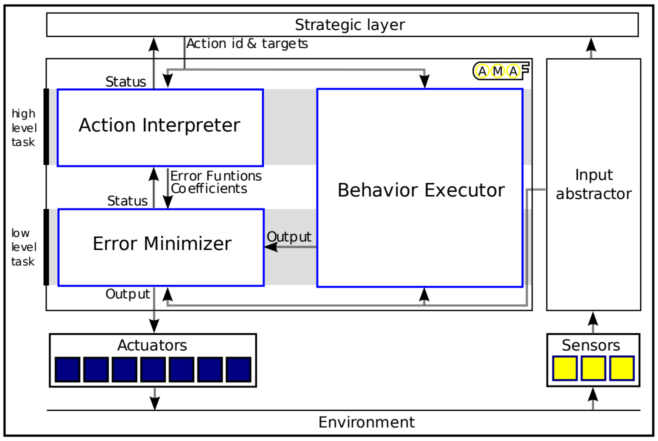


Fig. 10. Information flow of the performing modules of AMAF

structured into logs that stores all the information communicated by the performing modules and all the knowledge units with their scores. All the modules can read from all the logs of the Central Log System . This mechanism makes the communication inside of the framework flexible and permits the lack of synchronization between the performing modules and the learning ones.

AMAF works at two levels of abstraction (Fig. 10). At the high abstraction level, an action is expressed by an error function. At the low abstraction level, the action is expressed by the output vector that will be sent to the actuators.

These two divisions determine four subtasks each solved by a specific module. Finally there are two extra modules that are not necessary for the functioning of the framework but allow a significant improvement of the timing performance by modeling the behavior of the action through a direct mapping of the low-level action to the input data. For this motivation, these two modules work at the same time at both high and low abstraction levels.

- Action Interpreter learns at the high abstraction level. It has to transform the abstract action received from the strategy layer into an error function. If the action is the special action *train*, the Action Interpreter has even to decide, which action to execute between the ones defined by the Action Manager . During the execution modality, it has to determine the status of the abstract action.
- Action Manager learns at high abstraction level. It creates new actions and improves the learned ones. The score of an action indicates how interesting it is to execute that action during the training modality. The new actions are initially not sent to the robot. That way they can be executed only during the training phase. An action will be notified to the strategy layer only when its performances during the training are considered sufficient.
- Prediction Model Manager learns at the low abstraction level. It creates and updates the prediction models that will be used by the Error Minimizer to predict the effect of a low-level action. A prediction model is defined by a tuple $\langle P, \boldsymbol{M}, p, f \rangle$, where P is a subset of cardinality m of the perceived properties, $\boldsymbol{M}$ is a subset of cardinality n of output vector, and p is the predicted property. f is a function $f : \mathbb{R}^{m+n} \rightarrow \mathbb{R}$ that predicts the value that p will assume at the next input perception by knowing the actual values of a P and $\boldsymbol{M}$.
- Error Minimizer acts at the low abstraction level. It has to transform the error function in an output vector for each perceived input. The best output vector is the one that minimizes the error associated to the next received input. A time constraint can be set in order to compute the low-level action before the specified time interval.
- Behavior Model Manager learns to create and update the behavior models. A behaviour model is defined by a tuple $\langle P, \boldsymbol{M}, f \rangle$ where P is a subset of cardinality m of the perceived properties, $\boldsymbol{M}$ is a subset of cardinality n of the output vector and f is a function $f : \mathbb{R}^m \rightarrow \mathbb{R}^n$ that computes the values of $\boldsymbol{M}$ by knowing the values of P. Each model has to reproduce the behaviour of a step of an abstract action. Its score indicates the capacity to reach the success condition of the step.
- Behavior Executor acts by using the behavior model associated to the actual step of the abstract action in execution to directly compute the output vector. No search in the low-level action space is needed so the computational time is drastically reduced. The computed value is sent to the Error Minimizer that can use it for finding the expected best output configuration.

G. *Configuring AMAF*

AMAF can be tailored to the specific application through a few simple configuration possibilities:

- *Degrees of freedom*: the number of low-level actuators of the robot.
- *Controls*: AMAF requires the specification of the list of the controls used to generate the basic actions. Each of them can produce a basic action for each perceived property.
- *Modeling algorithms*: the creation of prediction models and of behavior models can be based on different modeling algorithms. AMAF requires at least one modeling algorithm. If more then one is specified, AMAF automatically chooses the modeling algorithm most appropriate for the current situation dependent on its prediction quality.

H. *Learning Flow*

Each learning module continuously generates the definitions of the new knowledge units, updates the definitions of the previously existing knowledge units, and generates an updated ranking of the defined elements. A ranking is a list of tuples $\langle id_q, s_q \rangle$ where id_q is the ID of an object q and $s_q \in \mathbb{R}^+$ its score.

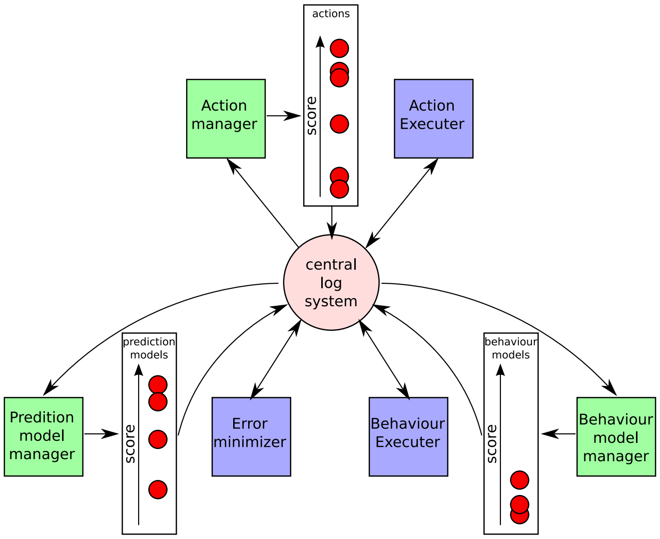


Fig. 11. Information flow of the learning process

The information flow of the leaning flow is shown in Fig. 11. The only constraint of the learning flow is that the definition of the knowledge units has to be compliant to the protocol specified by the framework.

For what concerns the information exchange, the CENTRAL LOG SYSTEM allows for a great variety in the implementation choices. The most trivial information exchange inside the learning flow is the one, in which the performing module uses the knowledge generated by the corresponding learning one.

The ERROR MINIMIZER has to continuously look inside the central log for new definitions of the prediction models and for updated rankings. Each time a perception arrives, the low level actions are evaluated by computing the expected future value of the properties controlled by the actual abstract action. The prediction models used to compute the expected values are the ones with higher score between the ones whose P is a subset of the actually perceived properties.

During the active modality, the ACTION INTERPRETER has to decide which action to execute. It is possible to exploit the ranking generated by the ACTION MANAGER by choosing the action with higher score.

Exploiting the knowledge generated by a learning module could be useful not only by the associated performing module but also by a different performing module or even by another learning module. For example, the BEHAVIOR MODEL MANAGER could build the behavior models by computing the expected best low-level action for different input configurations and then modeling the results. The capability of predicting the effect of the low-level actions would be supplied by the prediction models generated by the PREDICTION MODEL MANAGER .

I. *Performing Flow*

On the performing dimension, a fast and reliable information exchange is necessary in order to obtain a good reactivity. Therefore, it is not convenient to use a centralized system of the information exchange. AMAF manages the performing information flow by direct connections between the modules. The information flow is propagated from the higher abstraction level to the lower one. The work of each performing module can be monitored by reading its status.

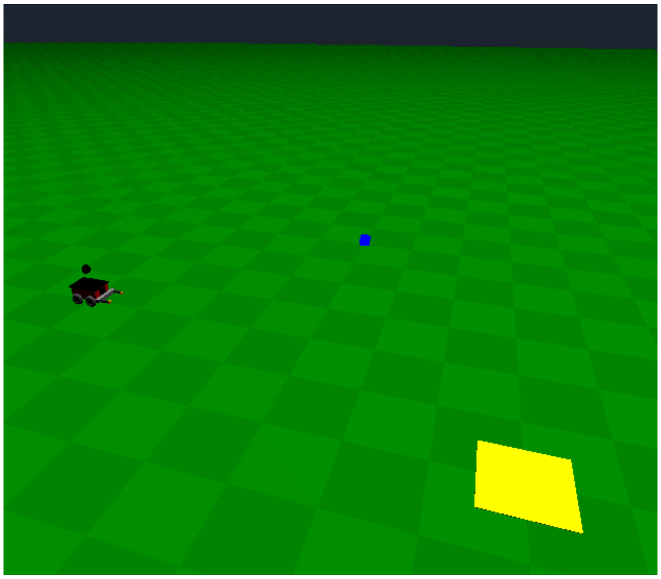


Fig. 12. Capture-The-Flag scenario. The robot has to learn to push the object to the yellow goal base. It has to learn by itself both the low-level actions and the strategy using them

V. EVALUATION

In this section, we will present the use of ESLAS in a Capture-The-Flag scenario. In the PlayerStage/Gazebo [16] simulation (Fig. 12) the well-known Pioneer2DX robot is used. The dynamics are simulated using the *Open Dynamics Engine (ODE)* [17]. This scenario consists of a goal base to which pucks dispersed in the environment have to be transported. The robot has to find out which skills, autonomously learned by the skill layer (Sec. III-C), have to be executed in which order, learned by the strategy layer (Sec. III-B), to achieve that goal. The results regarding the strategy layer are averages of 200 experiments, in which the robot had to push an object 30 times consecutively to the goal. The confidence interval of 95% is provided. The charts regarding the skill layer are individual examples.

The strategy's state space comprised the robot's relative angle and distance to goal g and object q:

$$(\alpha_g, d_g, \alpha_q, d_q) \in \mathbb{R}^4$$

Fig. 14 shows how the robot manages to abstract the 4-D state space into a small number of abstract regions, on which the actual strategy is learned.

The robot was equipped with only one drive as always stated. A positive lump-sum reward of 100 is given if the robot has pushed the puck to the yellow goal base. The change of the distance between the nearest puck and the goal is provided as reward rates. More formally, the motivation

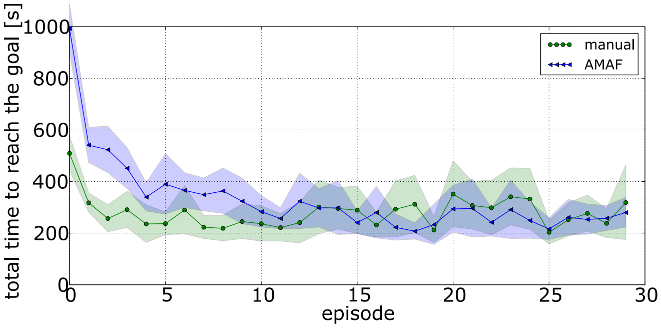


Fig. 13. Time to push the object to the goal

change was defined as follows:

$$\dot{\mu}_0(\alpha_g, d_g, \alpha_q, d_q) = \begin{cases} 100 & \text{if } d_g < 1m \\ \frac{d_q}{100} & \text{if } |\alpha_q| < 20^\circ \\ -0.01 & \text{otherwise} \end{cases}$$

The discount parameter of the strategy was set to $\beta = 0.1$. The state space adaptation heuristics, described in Sec. III-B.3, were parameterized as follows:

- Transition heuristics: $\theta_{TV} = 0.2$.
- Experience heuristic: The number of experiences was not bounded ($\theta_M = \infty$), but stayed below 10,000 (Fig. 14).
- Failure heuristic: $\theta_f = 0.01$.
- Reward heuristic: It considered the reward rates of the last $n = 20$ interactions made in the current region and used the constants $\theta_{RV} = 0.01$ and $\theta_l = 6$.
- Simplification heuristic: An action a in state s was considered deterministic, if $P(a|s) > 0.8$.

The configuration settings of the skill layer are described below.

- Degrees of freedom: 2
- Controls: “decrease” ($f_c(iv_p, av_p) = |av_p|$)
- Modeling algorithms: radial basis interpolation and polynomial approximation

The robot takes more time in the first run as it also has to explore its own capabilities and learn the skills, as can be seen in Fig. 13. From an average time of $1000s$ for the first episode the time needed drops quickly to slightly more than $200s$ (“learning”). Contrasted to that the “manual” curve displays the performance, which used the same strategy layer and same configuration, but replaced the learning skill layer by a handcrafted skill set of two optimal skills. It shows that while being faster in the beginning, the learning skill layer manages to finally converge to the same performance, which is assumed nearly optimal for this scenario.

The reward per second is displayed in Fig. 15, where the learning skill layer stays slightly below the optimal, but far less robust handcrafted skill set.

This shows that ESLAS is capable of autonomously tackle infinite state and action spaces in a realistic scenarios. Although the scenario was simple it showed all the characteristics of real-world scenarios, i.e. it was noisy, continuous, and time-dependent.

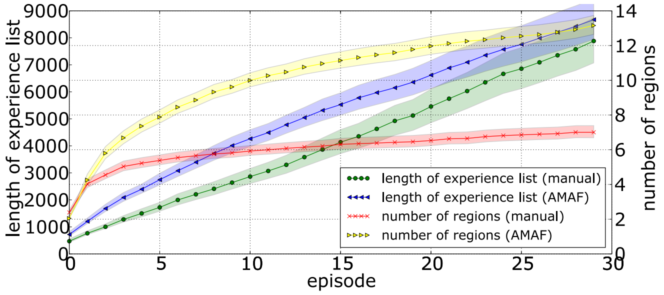


Fig. 14. Size of experience and number of abstract regions

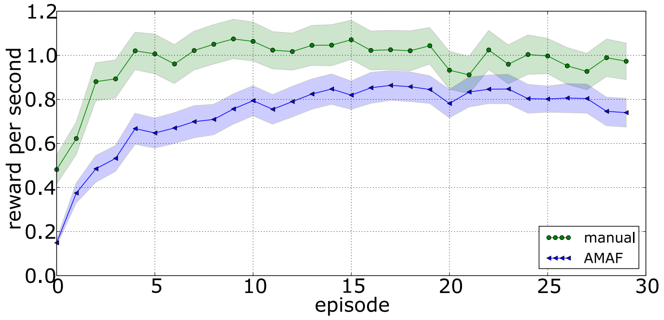


Fig. 15. The reward per second

The skill layer has autonomously generated different competing prediction models that determine the behavior of the learnt skills. We will try to represent in a synthetic way the behavior obtained by minimizing the angle to one object. We will use the prediction model based on the radial basis function approximation that predicts the next value of the angle by knowing the value of the angle and the distance to the object and the chosen low-level action. We have created a grid of 30x30 points in the input space. The input dimensions are the angle and the distance to the ball, so each point of the grid is characterized by a certain couple angle-distance. For each point of the grid, we have used the Error Minimizer to compute the low-level action that minimizes the predicted distance. The result is a pair of 3-D graphs, one indicating the chosen tangent speed and the other indicating the chosen rotation speed. We will represent the third dimension (the actuator intensity) by using colors: red for negative intensity, white for low intensity and blue for positive intensity.

In Fig. 16 the behavior of decreasing the angle to the ball is represented. The lower graph indicates the rotation speed. It is null when the angle is already minimized otherwise it is set to turn as much as possible toward the ball. The front speed is shown in the graph above and looks more confusing then the rotation speed. The only behavior that we can notice is that the robot goes back when it close to the object. In effect going on could lead to a continuous rotation around the object that would make the distance never decrease. When the distance is not low, there is not a clear behavior for what concerns the front speed. This makes sense because it does not affect much the angle to the object so it can even be chosen randomly without side effects on the performance of the action.

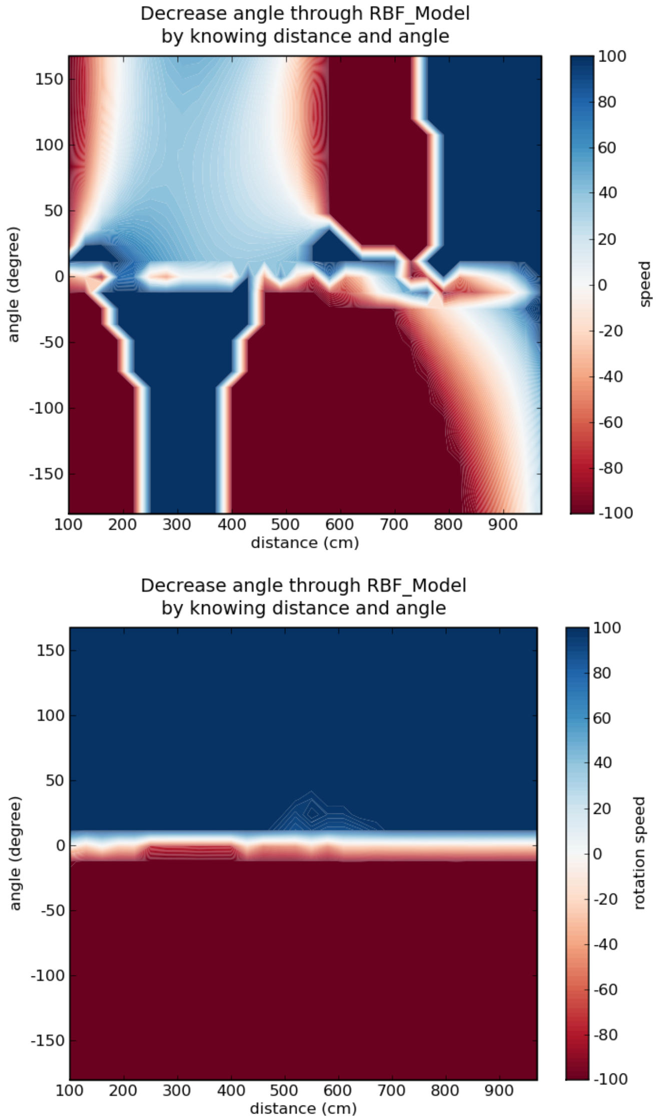


Fig. 16. Low-level actions associated to the abstract action of minimizing the angle to the ball. The red color denotes a full negative value (-100%), while the blue one a full positive one (100%).

VI. Conclusion

In this article we presented *Evolving Societies of Learning Autonomous Systems* (ESLAS), a framework that is able to handle system and environmental changes by learning autonomously at different levels of abstraction. It is able to do so in continuous and noisy environments by 1) an active strategy-learning module that uses reinforcement learning and 2) a dynamically adapting skill module that proactively explores the robot's own action capabilities and thereby provides actions to the strategy module. We presented results that show the feasibility of simultaneously learning low-level skills and high-level strategies while both are adjusting themselves to each other. Thereby, the robot drastically increases its overall autonomy.

This architecture is not only designed for individual learning robots, but also to support imitation in multi-robot scenarios as could be shown by the authors previous work [18], [19]. In the future, the authors are planning to use the ESLAS architecture also to enable robots to cooperate even if some of them were not specifically designed to do so. This means, that the robots will be able to detect behavior patterns in the performance of robots, which are not aware of the other robots around them. These patterns are then utilize to align the observing robot's own behavior accordingly.

References

[1] Willi Richert, Olaf Lüke, Bastian Nordmeyer, and Bernd Kleinjohann. Increasing the autonomy of mobile robots by on-line learning simultaneously at different levels of abstraction. In *International Conference on Autonomic and Autonomous Systems (ICAS'08)*. IEEE Computer Society, March 2008.

[2] Alexander Stoytchev. Five basic principles of developmental robotics. 2006.

[3] A. Stout, G.D Konidaris, and A.G. Barto. Intrinsically motivated reinforcement learning: A promising framework for developmental robot learning. In *The AAAI Spring Symposium on Developmental Robotics*, March 2005.

[4] H. van Hasselt and M.A. Wiering. Reinforcement learning in continuous action spaces. In *Approximate Dynamic Programming and Reinforcement Learning (ADPRL'07)*, pages 272–279, April 2007.

[5] Vijay R. Konda and John N. Tsitsiklis. On actor-critic algorithms. *SIAM J. Control Optim.*, 42(4):1143–1166, 2003.

[6] A. Lazaric, M. Restelli, and A. Bonarini. Reinforcement learning in continuous action spaces through sequential monte carlo methods. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 833–840, Cambridge, MA, 2008. MIT Press.

[7] A. Bonarini, A. Lazaric, and M. Restelli. Reinforcement learning in complex environments through multiple adaptive partitions. In *Proceedings of the 10th Congress of the Italian Association for Artificial Intelligence (AI*IA)*, pages 531–542, 2007.

[8] M. J. Kochenderfer. *Adaptive Modelling and Planning for Learning Intelligent Behaviour*. PhD thesis, School of Informatics, University of Edinburgh, 2006.

[9] Willi Richert and Bernd Kleinjohann. Adaptivity at every layer – a modular approach for evolving societies of learning autonomous systems. In *Proceedings of IEEE/ACM ICSE Workshop on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, Leipzig, Germany, 2008.

[10] T. M. Cover and P. E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13:21–27, 1967.

[11] A.W. Moore and C.G. Atkeson. Prioritized sweeping: Reinforcement learning with less data and less time. *Machine Learning*, 13(1):103–130, 1993.

[12] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley-Interscience, April 1994.

[13] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction.* MIT Press, Cambridge, 1998.

[14] C. J. C. H. Watkins. *Learning from Delayed Rewards*. PhD thesis, King's College, Cambridge, UK, 1989.

[15] Josep Call and Malinda Carpenter. Three sources of information in social learning. In K. Dautenhahn and C. Nehaniv, editors, *Imitation in animals and artifacts*, pages 211–228. MIT Press, Cambridge, MA, USA, 2002.

[16] Brian P. Gerkey, Richard T. Vaughan, and Andrew Howard. The player/stage project: Tools for multi-robot and distributed sensor systems. In *Proceedings of the International Conference on Advanced Robotics*, pages 317–323, Coimbra, Portugal, Jul 2003.

[17] Russell Smith. Website of ODE (Open Dynamics Engine). http://www.ode.org/, 2008.

[18] Willi Richert, Oliver Niehörster, and Markus Koch. Layered understanding for sporadic imitation in a multi-robot scenario. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'08)*, Nice, France, 2008.

[19] Willi Richert, Ulrich Scheller, Markus Koch, Bernd Kleinjohann, and Claudius Stern. Increasing the autonomy of mobile robots by imitation in multi-robot scenarios. In *International Conference on Autonomic and Autonomous Systems (ICAS'09)*, 2009.

User-centric Identity Management in Ambient Environments

Hasan Ibne Akram
Fraunhofer Institute for Secure Information Technology
Munich, Germany
hasan.akram@sit.fraunhofer.de

Mario Hoffmann
Fraunhofer Institute for Secure Information Technology
Munich, Germany
mario.hoffmann@sit.fraunhofer.de

Abstract– **Context-aware intelligent systems in ambient environments will have major impact in the near future to the way people will perceive and deal with computer technologies regarding privacy, security and trust. In those environments it will be all about personalized information and digital identities – so the foremost goal we are heading for in our research is: How to avoid *omni-persistence* in a world of *omni-presence*?**

Firstly, we show in this paper how any kind of personalized information, such as identities, preferences and profiles, will fuel those systems to support, serve and simplify people's lives. Secondly, we are convinced that especially privacy and so-called informational self-determination are at stake if protection goals like confidentiality, transparency, and minimal disclose of information are not well balanced and precisely taken into account when realizing such systems.

Existing standards, solutions and technologies in Identity Management are specifically tailored for example for the Internet, company processes or eGovernment. However, for future ambient environments they have to be improved and revised to meet also user-centric requirements. This paper combines certain aspects of existing approaches to introduce a new middleware architecture that supports user-centric Identity Management. We further show that this middleware enables future application developers to meet (almost) all of our postulated ten laws of identity.

Keywords - Identity Management, Ambient Environments, Privacy by Design, Identity Metasystem, Higgins

I. INTRODUCTION

Inhabitants of ambient environments are envisioned to be surrounded by smart devices that are working continuously to make their lives more comfortable. This is achieved by context-aware intelligent systems where virtual networks (converging fixed, wireless and mobile) consist of numerous nodes, smart devices, sensors and actuators. The underlying system is transparent but omni-present to the users and the users are omni-present to the system. The basic research area is known as Pervasive and Ubiquitous Computing (a synonymous term widely used in Europe is Ambient Intelligence).

The basic idea is: The more information about the inhabitants of such environments is fed into the context aware systems working in the background, the better or more personalized it works for them. At the same time, however, it has to be ensured that the inhabitants' privacy is not endangered in such smart environments. The fact that depending on the application area context aware systems in principle are able to store and aggregate whatever information about individuals, groups and communities has to be taken into account seriously. Omni-presence shall not lead to omni-persistence. The most important questions are

- 'What information is stored, aggregated and mined?',
- 'Who is authorized to get access to such information?', and
- 'How long will the information being stored?'.

Thus, privacy and context awareness in smart environments, although being rather contradictory issues, have to be put in practice in a balanced manner. Therefore, in this paper an inherently secured user-centric Identity Management framework is proposed that deals with the *complete life cycle of identities* of users, services, and devices as well as users' awareness in information disclosure and privacy.

This paper elaborats step by step an architecture for an Identity Management Solution for such scenarios. Firstly a typical scenario for ambient environment is shown followed by a brief description of the ten laws of identity for ambient environments which have been previously discussed in [12]. A study on the state of the art technologies and an evaluation based on the ten laws of identity is presented in the following. Finally, based on the evaluation we propose an architecture for Identity Management in ambient environments that is compliant to (almost) all the ten laws of identity.

II. IDENTITY IN AMBIENT SCENARIOS

In the literature many kinds of future application scenarios which may benefit from the support of context-aware smart environments have been introduced already. Examples are intelligent buildings, automotive, and healthcare. In order to illustrate the most typical user-centric requirements we will, therefore, focus on a typical test scenario taken from an EU project for ambient environments called Hydra[1] [6, 7] (the authors are part of the consortium).

[1] Hydra: Networked embedded system middleware for heterogeneous physical devices in a distributed architecture. http://www.hydramiddleware.eu (2007) contract number: IST-2005-034891, duration: 07/2006-06/2010.

In the second section of this chapter we will then summarize the ten laws of identity that we have defined in our previous paper [12]. There you can find a detailed description and analysis with respect to the scenario sketched below. The ten laws of identity then serve as the basis of the architecture discussion and evaluation in Chapter V.

A Scenario Definition

In Hydra, fictitious scenarios have been derived in three domains: Building automation, healthcare, and agriculture, which are likely to be practiced in reality in 2015 [7, 8]. Many of these scenarios are derived from business cases from the perspective of an end-user; i.e., from application level. As a consequence, Identity Management can have a large range of implications to information systems encompassing role-based access control, *Single Sign On (SSO)* in single and cross organizational domains, as well as management of virtual identities, identity life cycles and sessions. However, in case of designing a middleware for Identity Management the perspective of requirements analysis shifts from the end-user to a developer. The question is, thus, which requirements coming from application domains can and should be addressed in a middleware?

With the intention to illustrate the necessity of an Identity Management System in a middleware for developing ambient applications we will take as a basis a detailed technical scenario of a heating system breakdown at "Krøyers Plads" housing complex located in Copenhagen that deploys the "Hydra Building Automation System" (HBAS) [7]. The resident living in a new flat in this building complex is equipped with automated lamps, computers and a wireless network, as well as a Hydra-enabled heating system and many other usual sets of integrated embedded devices. While the resident is at his office, the heating system of the flat breaks down and the water pressure rapidly decreases down to a level that is detected as an emergency situation by the HBAS which is shown as legend 1 in Figure 1. As a result of that HBAS sends out an alert message to the resident (legend 2 in Figure 1).

In order to get the heating system fixed as soon as possible the resident chooses a service provider from a list of providers matching the emergency requirements and his preferences best. The service provider then sends a service agent (e.g., a specialized technician) to the house. The major challenge here is to allow remotely a particularly authorized service provider and his technician to get into the house to fulfill a specific task. Therefore, included in the repair order a dedicated and restricted HBAS authorization ticket guarantees that in this case a service agent can enter the flat and get access to the heating system (legend 3 and 4 in Figure 1). After entering the flat upon successful authentication procedure the service agent gets authorization to access additional context aware information required to perform his job (legend 4 in Figure 1).

This representative scenario can be basically adopted by many kinds of similar scenarios of remote authorization such as large housing areas with housekeeping service, office buildings with restricted access, airports, and hospitals. Thus, with the basic scenario of Hydra being illustrated we can go one step forward in our process of our identity requirements analysis in ambient environments.

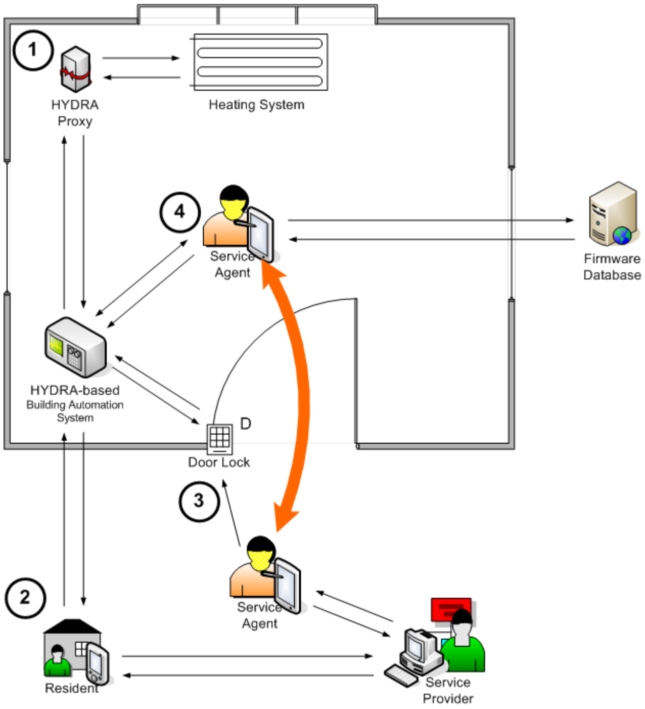


Figure 1: Sequence of steps for the technical scenario [7]

In our previous paper [12] we showed an extended use case analysis of the given scenario, applied the principles of *Federated Identity* in the process of use case analysis and derived ten laws of identity for ambient environments. The details of the use case analysis process can be found in the referenced paper. In the next subsection we will briefly describe the ten laws of identity and their implications in ambient environments.

B Ten Laws of Identity

Identity Management in ambient environments, characterized by pervasive and ubiquitous computing, has been explored by researchers intensively during the last decade. Requirements and principles of Identity Management have been analyzed and derived based on certain needs in certain scenarios. A prominent example is the "Laws of Identity" by Kim Cameron tailored for the Internet [1].

Obviously, these related works have some commonalities and disparities among themselves. This is simply because all these laws and requirements are based on some variable parameters; namely - perspective, time, computing environment etc. Therefore, there is a need of customized adoption and modification of the existing laws to certain scenarios. In this section we postulate the following ten laws of identity which are meant for ambient scenarios as shown in Section II.A:

1. User Empowerment: Awareness and Control
2. Minimal information disclosure for a constraint use
3. Non-repudiation
4. Support of directional identity topologies
5. Universal Identity Bus
6. Provision of defining strength of identity
7. Decoupling Identity Management layer from application layer
8. Usability issue concerning identity selection and disclosure
9. Consistent experience across contexts
10. Scalability

1) User Empowerment: Awareness and Control

Our first law looks similar to Kim Cameron's first law of identity where it says *"User Control and Consent"* [1]. We do totally agree that user consent and control are necessities in Identity Management but at the same time we believe that the word "consent" does not imply a total *empowerment* of the user. According to the definition of the word "consent" provided in American Heritage Dictionary[2], is - "To give assent, as to the proposal of another; agree." This merely implies an agreement and nothing beyond an agreement; i.e., it does not imply that the user being fully aware of the consequences of the agreement. The following example of one of today's extremely popular Web 2.0 applications examines why a mere agreement of the user is not enough. The "Contact importer" feature of facebook.com has been a very much well-liked feature and it has been very trendy in many other web 2.0 applications.

Figure 2 shows a screen shot of facebook's contact importer feature. Using this feature a user is able to import the user's buddy list from his other email or instant messenger accounts like Google, GMX, MSN, Yahoo, AOL, and many others. What the user has to do here is to provide his username and password credential to facebook and facebook uses that credential to import the buddy list from the corresponding provider. This allows facebook to have access to all the other accounts of the user and even if we consider facebook as a basically trusted party, privacy of the user has been completely compromised.

In this example the username and password have not been stolen without the user's consent, i.e., the user had agreed to giving his username and password and clicked the "Find Friends" button. The question to ask would be if the user is *aware* of the fact that his privacy has been compromised. Therefore, instead of "consent" the first law of identity takes the word "awareness" which subsumes "consent" anyway.

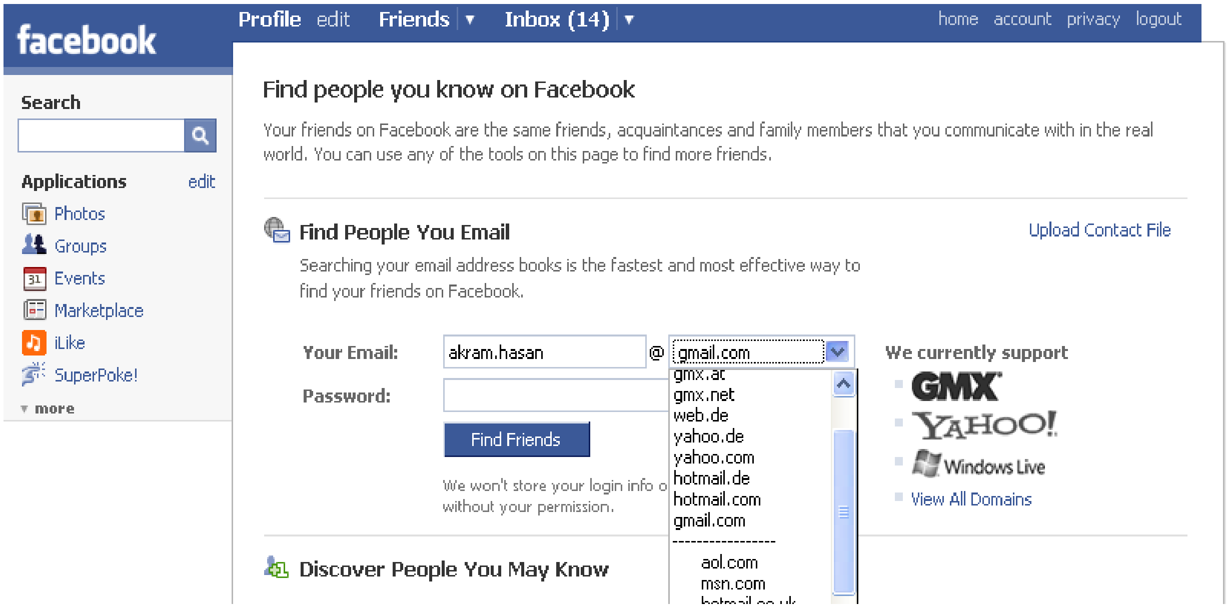
From the perspective of our scenario (Section II.A) the first identity requirement concerns the user in an ambient environment and emphasizes on two key words – "awareness" and "control". In a transaction taking place between two entities in such ubiquitous scenarios, each entity must have full knowledge regarding the information he is about to disclose and to whom he is about to disclose. Besides having full knowledge about the information disclosure the entities must also have full range of control power to decide whether to disclose a particular set of information or not [1] as well as the power to continuously check the authenticity of this information and even change or delete it.

2) Minimal Information Disclosure for a Constrained Use

Whereas the first law has addressed awareness and control, the second law addresses information disclosure. Basically these two laws are complimentary to each other. In a ubiquitous scenario there can be numerous possible ways information can be leaked out without the user being aware of the information disclosure. Therefore, the system must ensure that claims must be satisfied with a minimum set of information required. The support of zero-knowledge-proofs for example is favored over disclosing a credential.

From the perspective of our building automation scenario (Section II.A) the second law of identity means the following: We have already stated that there is a contractual relationship between the resident and the service provider. Therefore, authentication information propagates in a transitive fashion to the service agent; i.e., since the agent is authenticated by the service provider, he is also authenticated by the resident and depending on the security policy all or parts of the smart devices in his apartment. In the process of fixing the heating system, the service agent will need to have access to certain information, e.g., the usage pattern of the heating system. Here the service agent must be provided with a minimal information set that is only relevant for fixing the heating system. The usage pattern of the heating system supplied by the smart devices to the service agent must somehow guarantee that no other information is retrievable from it that goes beyond the necessity of fixing the heating system, e.g., the service agent should not be able to figure out from the usage pattern that during which period of the year the resident is on holiday or remains out of the flat.

[2] Consent. (n.d.). *The American Heritage® Dictionary of the English Language, Fourth Edition*. Retrieved July 03, 2008, from Dictionary.com website: http://dictionary.reference.com/browse/Consent



Find people you know on Facebook

Your friends on Facebook are the same friends, acquaintances and family members that you communicate with in the real world. You can use any of the tools on this page to find more friends.

Find People You Email

Searching your email address books is the fastest and most effective way to find your friends on Facebook.

Your Email: akram.hasan @ gmail.com

Password:

We won't store your login info o without your permission.

Discover People You May Know

Figure 2: A screenshot of facebook's contact importer feature (screen shot taken on 5th September 2008).

3) *Non-repudiation*

The term "Non-repudiation" has a traditional legal meaning and at the same time, a different meaning in terms of digital security [19]. We will focus on the latter meaning of "Non-repudiation" and then relate its necessity to our scenario (Section II.A). In a crypto-technical sense transfer of data from one entity to another must guarantee authenticity, integrity, and a time stamp, so that neither of the parties involved can deny that the transfer of the data took place.

Within the scope of the building automation scenario (Section II.A) the issue of authenticity takes place in the following process: The endpoint of the service provider receiving a message from the endpoint of the resident must know whether the message is really transmitted from the resident or if it is under a spoofing or masquerade attack [4]. Therefore, there is a need of mechanism(s) that guarantees identity preservation.

In order to illustrate integrity, we continue with our running scenario example: The service provider receives a message from the resident over HTTP, he must guarantee the integrity of the message content. From a middleware viewpoint, there must be supports that allow the developer ensuring that the messages sent from one node to another is not being modified or misused in an intermediary node or is not under falsification attack [4]. In order to guarantee integrity it is also important that any kind of message manipulation has to be detected.

Another vital point is to ensure that a time stamp is attached to the message. This is required to combat replay attacks. A time stamp attached to the message will make the message valid only for a certain period of time and as a result of that will lower the probability of replay attacks.

Summarizing, unforgeable identity, non-falsifiable message exchange, and provision of a time stamp are required in middleware for such scenario so that the identity of the sender and the integrity of the message cannot subsequently be refuted.

4) *Support of directional identity topologies*

Kim Cameron identified identity as a vector rather than a scalar in his paper [1], i.e., identity not only has magnitude but also a direction. In his fourth law of identity he expressed the need of omni-directional and unidirectional identities. We have adopted this law in the context of pervasive computing and have modified it according to ambient environments' needs.

In the domain of ubiquitous computing, communication takes place in various topologies and so does *identity federation*. Identity federation in such scenarios can be unidirectional, bi-directional or even omni-directional. An unidirectional federation involves an Identity Provider (IdP) issuing a Security Token for a user when a particular Relying Party (RP), e.g., a service provider, the user wants to get access to, is asking for it. Bi-directional federation takes one step further, where the RP is able to act as an IdP once the user is federated to the RP by an IdP within the circle of trust. This is how authentication information is being propagated node to node [12] in our home automation scenario (Section II.A). Finally, an omni-directional identity refers to a virtual identity emitted to any entity that shows up. An example with respect to our scenario would be, the presence of the service agent is being sensed by the intelligent devices at the resident's apartment when the service agent transmits his identity in omni-directional manner.

The fourth law of identity states that the following identity federation topologies must be supported in an Identity Management System in ambient environment:

1. Broadcast (omni-directional)
2. Point to point (unidirectional or bi-directional)
3. Multicast (omni-directional and/or bi-directional).

5) *Universal Identity Bus*

In today's Internet users have multiple virtual personas for one identity and each of these multiple personas has different contexts, purposes and flavours. In the world of *Internet of Things* it can well be imagined that these multiple personas would require being portable from domain to domain, device to device or context to context. No portability of identity will create *Identity Silos* and cross domain interoperability or even inter domain interoperability across platforms or devices will be challenged.

The middleware for an ambient environment Identity Management System inherently requires supporting interoperability between the garden varieties of Identity Management technologies available from different vendors. The fifth law of identity for ambient environments states the necessity of a Universal Identity Bus (UIB) that will provide vendor to vendor interoperability functionalities. In order to achieve this requirement the middleware must support UIB that works as a bridge between different Identity Management technologies.

6) *Provision of defining strength of Identity*

In order to illustrate why such ambient environments necessitate the provision of the strength of identities two aspects have to be taken into consideration: identity propagation and the dependency of the identity.

Identity Propagation: In a federated environment identity can be lightweight or rather strong depending on policies of the IdP and the RP. Especially, when it comes to bi-directional federation (see law 4) it is important to categorize identity according to its strength. In such federation RP is gaining the power to be the IdP once an entity is authenticated to it by the original IdP and the propagation of identity can continue creating a very long chain in ubiquitous computing which may result in an apocryphal identity. Thus, there is a need of accumulated calculation of its source of identity reliability.

Dependency of the Identity: In a ubiquitous world virtual identities might refer to individuals, devices or services, i.e., more general entities or things. If a device is owned by a person, for example, the identity of the device is somewhat depending on the identity of the person, i.e., the identity of the device is incomplete without relating it to an identity of another entity. In a similar way many use cases may arrive where an identity does not suffice itself without being depending on an identity of another entity. Based on this criteria identity can be categorized to be strong (independent), weak (dependent) or somewhere in the middle. Thus, we can justify the requirement of a provision of having the strength of an identity in the middleware. It is important to note that weak identities and strong identities are not the same as sub-identities, which are basically subsets of identities. Identities or sub-identities both can be rated by their strength depending on their degree of being autonomous.

7) *Decoupling Identity Management layer from application layer*

This requirement builds up another block on top of the "*Universal Identity Bus*" and separates the application layer from the Identity Management Layer. This is obligatory for the our Identity Manager for two main reasons: 1) organizations are being able to change their identity policies without having an impact on the business layer and 2) the developers have an environment where they can work on the identity layer being transparent of the business layer or vice versa.

8) *Usability issue concerning identity selection and disclosure*

"A potato peeler is easier to use for peeling potatoes than a knife is, but a lot harder to use for murder." – Ross Anderson [15]

The above quotation figuratively expresses the fact that usability is case specific. High usability of a tool in a certain area can be extremely inconvenient for other purposes. Therefore, appropriate design support of usability for identity selection and discloser is unavoidably important in a middleware.

We have already emphasized the issue of empowerment of the user in case of revealing information in our first identity requirement. Lack of usability will make law 1 (User Empowerment: Awareness and Control) almost impossible to take place. In a user-centric design the user is the ultimate procurer and a methodic requirement specification of usability keeping the procurer in mind is unavoidable [10]. Therefore, our middleware architecture must facilitate the developer with adequate support for implementing usability.

9) *Consistent experience across contexts*

Context is one of the major concerns in our test scenario (Section II.A) and identity and context are closely related. Therefore, while analyzing requirements of Identity Management in ambient scenarios, the issue of context is considered. In ambient environments an entity and its identity will have an *n* to *m* relationship, i.e. one entity (e.g., a user, device or service) can have multiple identities and one identity can be possessed by several entities. For example, the resident has several identical sets of devices, e.g., temperature or movement sensors, and he wants to use them with one single device identity. In this example one identity is shared by multiple entities. The example one entity having multiple identities would be, the resident has an identity at his work, a different one for his shopping web sites and another different one for heating system repairing service providers. So identities may change in different contexts based on different roles. In this *n:m* relationship of identities and entities it is very important to have consistence experience for the user depending on contexts.

Along with the consistencies among context, the identities provided in different contexts should also be independent of each other, i.e., the identity the user provides at work should not be related to his identity for his shopping website and vice versa. This is in order to avoid aggregation and concatenation of partial identities following the principle of privacy by design.

10) Scalability

Identity multiplies with time. For the inhabitants of ambient environments a growing number of identities across contexts must be managed properly and at the same time there has to be room for conceiving new identities. Moreover, in an ambient environment the number of nodes joining in and out is dynamic and thus, the capability of an identity to interacting with the identities of the other numerous nodes is necessary. Therefore, scalability of identity refers to an entity that must be able to spawn new identities and a single identity must have the capability to communicate with a growing number of identities.

III. ARCHITECTURAL IMPLICATIONS

Having the laws of identity being illustrated in the previous section we will get back to our home automation scenario (Section II.A) in order to motivate our architectural approach and to analyse the implications. In this section we will see use cases in the home automation scenario (Section II.A) where the propagation of authentication information from entity to entity is based on contractual relationships. We will also observe how the three roles of *Identity Federation – Subject*, IdP and RP – shift from endpoint to endpoint.

A Use case analysis

The first identity federation use case in our scenario is shown in Figure 3. In step 1 the resident is sending a request to the service provider for a service agent to be sent to his flat to fix his heating system. In step 2 the service provider is asking for his credential as a set of claims. Here the resident has an option to choose an IdP that can satisfy the claims from the RP that happens to be the service provider in this case. For simplicity we assume that the resident himself is able to issue an identity token that satisfies the claims and would also be accepted by the RP. So, in step 3 the resident issues himself a token and in step 4 releases it to the service provider. After receiving this token the service provider issues a co-signed token to a service agent (step 5) who is to be sent to the resident's flat for repairing the heating system.

Another use case scenario is shown in Figure 4. Here, the service agent has to authenticate himself at the door lock of the resident's apartment. The roles - *Subject*, RP, and IdP – have been shifted to the service agent, the door lock, and the service provider correspondingly. In step 1 the service agent sends a request to the door lock for accessing the appartment. In step 2 the door lock sends a request for a security token as a set of claims. The service agent requests his IdP (the service provider in this case) for a security token satisfying the claims. The service provider issues a token in step 4 and in step 5 the service agent releases this token to the door lock.

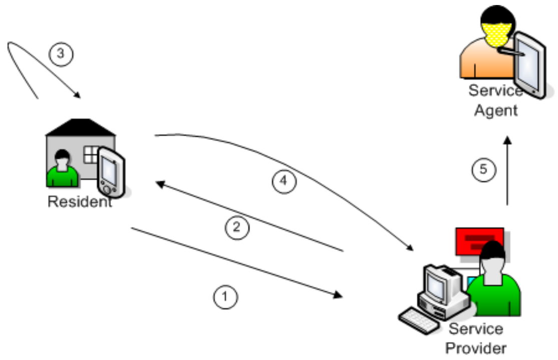


Figure 3: Sequences in the process of the resident authenticating himself to the service provider and the service provider issues a cosigned token to the service agent.

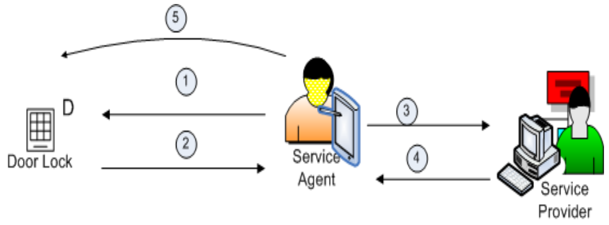


Figure 4: Sequences in the process of the service agent getting authenticated by the door lock of the resident.

Due to transitivity of authentication information flow as a part of the contract between the resident and the service provider, the door lock accepts his request; i.e., the door lock accepts the authentication assertion (in form of a security token) from the resident, the resident sends the token to the service provider and the service provider issues a co-signed token to the service agent, consequently the door lock accepts the authentication information of the service agent. This process is repeated in each identity discovery taking place in the scenario.

B Distributed Identity Provider

The use case analysis clearly motivates us toward the concept of *Distributed Identity Provider.* When bi-directional federation takes place in a way that every endpoint in the circle of trust[3] (or contractual relationship) can attain the role of IdP and RP the set of the endpoints collectively can be defined as *Distributed Identity Provider*. In Section V we will elaborate in details where the architecture is illustrated. The following section is dedicated to state of the art analysis that evaluates the suitability of the existing IdM technologies to fabricate Identity Management architecture for ambient environments.

IV. STATE OF THE ART

This section will accomplish a state of the art study of standards, frameworks, protocols, and products related to Identity Management. For these purposes we apply our ten laws of identity illustrated in Section II.A in order to find out the closest technology which complies with the corresponding requirements. Therefore, we start with a subsumption of basic enabling standards from the WS-* family and dedicate the following sections to state of the art technologies, namely SAML, OpenID, Windows CardSpace, Higgins, and Liberty Alliance.

A Web Service Related Standards

Identity as a service is a visionary goal of the Service Oriented Architecture (SOA) proponents. The adoption of the spirit of identity as services in futuristic ambient computing is also being pushed by the researchers and scientists. Since Web Services is considered as being a key driving technology for enabling SOA we will briefly highlight some Web Service related standards that are as well relevant for an Identity Management ecosystem.

B WS-Security

The main objective of WS-Security is to secure the Web Service message itself. The SOAP message is secured in order to guarantee authenticity, integrity, and confidentiality of the message. Moreover, it also provides a time stamp for SOAP messages [4].

WS-Security is relevant in our IdM architecture because this standard provides support for our third law of identity (Non-repudiation). Thus, WS-Security is considered as a candidate for being one of the building blocks of the architecture.

C WS-Trust

WS-Trust defines a framework that provides protocol agnostic ways to issue, renew, and validate security tokens. Moreover, it defines ways to establish, assess the presence of and broker trust relationships. The main goal of WS-Trust is enabling applications to construct trusted SOAP message exchanges [11].

The reason WS-Trust is interesting or relevant in our ambient scenario (Section II.A) is, in the home automation scenario, we have seen that the contractual relationship between the service provider and the resident needs to be somehow technologically represented. WS-trust exactly addresses this issue. Moreover, being agnostic to security tokens, the support of WS-Trust in the architecture enables the developer to take advantage of any kind of associated protocol.

D WS-Policy

WS-Policy is a language for representing capabilities and requirements of a Web Service. In other words, it tells the consumer of a Web Service what the requirements are that it must fulfill in order to consume that service. These requirements can also be optional in some cases, which would provide certain advantages to the client if it can fulfill those optional requirements [9]. WS-Policy provides a precise way to write policy expressions for a certain Web Service.

Getting back to law seven (Decoupling Identity Management layer from application layer) we can clearly relate WS-Policy to our requirements. In order to achieve such goal, changes in identity policies should not affect the business policies or vice versa. WS-Policy offers functionalities to facilitate such mechanisms.

E WS-Security Policy

WS-Security Policy language is built on top of the WS–Policy framework and defines a set of policy assertions that can be used in defining individual security requirements or constraints. The motivation for adopting WS-Security Policy in our architecture is the same as for adopting WS-Policy.

F WS-Federation

WS-Security, WS-Trust, and WS-Policy/WS-SecurityPolicy described in the previous sections provide a basic model for federation between IdP and RP. WS-Federation uses these building blocks to define additional federation mechanisms that extend these specifications and leverage other WS-* specifications [21].

WS-Federation allows security realms to broker identities, user attributes and authentication between Web services. This is an essential factor for engineering a *Distributed Identity Provider* architecture.

G WS-MetadataExchange

Web Services use Metadata to describe what other endpoints need to know to interact with them. For example, WS-Policy describes the capabilities, requirements, and general characteristics of Web Services; WSDL describes abstract message operations, concrete network protocols, and endpoint addresses used by Web Services; XML Schema describes the structure

[3] According to the definition stated in OASIS standard: *Web Services Security: SOAP Message Security 1.0* [11] - "Trust is the characteristic that one entity is willing to rely upon a second entity to execute a set of actions and/or to make a set of assertions about a set of subjects and/or scopes."

and contents of XML-based messages received and sent by Web Services [20].

In order to bootstrap communication with a Web Service, this specification defines how an endpoint can request the various types of Metadata it may need to effectively communicate with the Web Service.

H IdM Protocols & Technologies

1) OpenID

OpenID 1.0 was originally developed in 2005 by Brad Fitzpatrick, Chief Architect of Six Apart, Ltd. It is now set up by a wide range of websites, especially which have heavy user-generated contents. OpenID Authentication 2.0 [13] is now turning into an open community-driven platform that permits and motivates federated identity. And the community is on its way for preparing drafts of a fully backward-compatible OpenID Authentication 2.0 specification which is a data transfer protocol to support both push as well as pull use cases. Besides, the community is coming up with extensions to support the exchange of rich profile data and user-to-user messaging [3].

According to an article published in German online computer magazine “Heise Online[4]” on 18th January 2008 there exist already 370 million OpenIDs globally. However, the number of really *active* OpenID users is still unknown. Big companies like Yahoo, AOL offered an OpenID to all their users and as a result, the number of existing OpenID naturally jumped up to such a high number.

There are three key features of OpenID: Single Sign On, decentralized, and light weight identity.

Vulnerabilities of OpenID:

Firstly, OpenID also allows the RP to redirect the client to the IdP for authentication at the IdP site [22, 13]. Therefore, it raises the probability of phishing. The user has no control over choosing his *Identity Provider* and therefore the first law (User Empowerment: Awareness and Control) of identity is violated. The second problem with OpenID is that the URL that is used to identify the *Subject,* is recyclable. Since OpenID permits URL based identification, it brings the issue of privacy. The privacy of the user using an URL as his OpenID would be compromised if somehow lost the possession of that URL.

2) SAML

The most precise and shortest way of defining SAML, presented by Eve Maler, is: “*The Security Assertion Mark-up Language in six words:* ***The universal solvent of identity information.”*** SAML comes with the spirit of portable identity.

SAML (Security Assertion Markup Language) is developed by the OASIS Security Services Technical Committee with an objective of conveying security information across cross-organizational boundaries. There are three official versions of SAML – SAML 1.0 was the first official version coming out in November 2002, it was followed by SAML 1.1 in September 2003 and the latest version: SAML 2.0 has come out in March 2005 [24, 25].

Vulnerabilities of SAML:

SAML can be configured in a very lightweight (less secured) identity way and at the same time it can be configured in a much secured manner. In SAML an assertion is a set of security information that is requested by a *RP* about a particular *Subject* or entity. IdPs transport assertions to RPs who allow the requests. In the Google Single Sign On (SSO) implementation, the authentication response did not include the identifier of the authentication request or the identity of the recipient [23]. This may allow malicious RPs to impersonate a user at other RPs.

3) Liberty Alliance Project

Liberty Alliance started its expedition in 2001 with the purpose to be the service provider of the open standards organization for federated Identity Management. Guaranteeing interoperability, supporting privacy, promoting adoption for its specifications, providing guidelines and best practices Liberty Alliance has the objectives to enable users to protect their privacy and identity, to enable SPs to manage their clients lists, to provide an open federated SSO, and to architect a network identity infrastructure that is compatible with all emerging network access devices [17].

Vulnerabilities of Liberty:

Liberty Alliance technology stream is mainly based on SAML 2.0 and therefore inherently it suffers from the similar vulnerabilities as SAML stated in the previous section.

4) Windows CardSpace

Windows CardSpace is a visual metaphor for identity selectors for the end-user. Windows CardSpace provides controlling power to the end-users on the fact which information (about the end-users) should reach to the *Relying Party* and which should not. Windows CardSpace is a production of Microsoft shipped with Windows Vista (or as an add-on in Windows XP); it is not meant to replace the other standards handling digital identity rather to utilize and extend them [2]. Windows CardSpace is token agnostic.

The limitation and criticism of CardSpace is – although it does support virtually any security token format, it is not protocol agnostic. Currently it is only compatible with the WS-* Web Services protocols, which center on WS-Trust. For the reason that it is token

[4] http://www.heise.de/security/Yahoo-will-das-Passwort-Chaos-beenden--/news/meldung/102001

agnostic, but tied to WS-* protocols, we can say that it only partially complies with the fifth law which postulates the need for protocol agnostic as well as token agnostic (Universal Identity Bus).

Vulnerabilities of Windows CardSpace:

On top of its limitations CardSpace has some flaws: Firstly it relies on the users' judgements on the trustworthiness of Relying parties (RPs). A CardSpace user is given the freedom to choose one of the options of high-assurance certificates belonging to the RP, ordinary certificates belonging to the RP or RP with no certificates [14]. In terms of the first law (User Empowerment) this certainly gives a lot of power to the user. At the same time the option of allowing RP with no certificates weakens the compliance with the third law (non-repudiation).

The second vulnerability is, Windows CardSpace relies on a single layer of authentication. The user has to be authenticated to the IdP using traditional authentication mechanisms. If a working session is somehow hijacked or the password is cracked, the security of the whole system is compromised. This has been practically shown by two IT-Security students at Horst Görtz Institute for IT Security (HGI), Bochum, Germany, where they manipulated the DSN server to implement a dynamic pharming attack [18].

5) Higgins

Higgins is a software infrastructure that supports consistence user experience that works with digital identity protocols, e.g., WS-Trust, OpenID, SAML, XDI, LDAP etc. The main objectives of the Higgins project are the management of multiple contexts, interoperability, and the definition of common interfaces for an identity system. Various technologies including LDAP, SAML, WS-*, OpenID etc. can be plugged into the Higgins framework.

The first version, Higgins 1.0 was released in February 2008. The next version, Higgins 1.1 is supposed to be released by June 2009. There are also ideas and concepts in discussion beyond Higgins 1.1.

The architecture of Higgins 1.0 is based on:

- An Identity Attribute Service (IdAS): It provides a virtualized, unified view and a common means of access to identity information from multiple heterogeneous data sources. Simultaneously supports multiple Context Providers to abstract identity information from LDAP, SAML, OpenID, InfoCard, RDF.
- An infocard provider and Security Token Service (STS): It uses IdAS in a way that identity information comes from multiple Identity Providers.
- Multiple forms of Identity Agents: Web-based and client-side card managers are supported as well as browser extensions, and user interfaces (InfoCard selectors).

In Higgins 1.1 it is expected that an enhanced InfoCard with additional features will be supported. Among the enhanced InfoCards two very promising ones are *z-cards* and *r-cards*. The z-card adds functionalities to the managed card (*m-card*). It offers more privacy by caching the security token locally, and it supports subsets of claims. It also supports zero-knowledge proofs, thus enhancing privacy and trust features. An r-card is an enhanced version of managed cards (m-cards) and personal cards (*p-cards*). It sets up a data synchronization relationship between the user and the *Relying Party*. A change at either side updates the other.

Information cards created in CardSpace can be used in Higgins but the z-cards and r-cards created in Higgins are not currently supported in CardSpace. Both systems are in their early stages, and changes in compatibility are expected as this high-level identity architecture catches on.

The ideas and concepts in discussion beyond Higgins 1.1 are targeting the mobile platform which may be named as "Mobile Higgins". The target platforms are Symbian, RIM, Windows, Mobile 6, iPhone, Android etc.

Vulnerabilities of Higgins:

Since Higgins supports various IdM protocols and technologies it inherently takes over the flaws and vulnerabilities of those technologies and protocols. It also does not provide supports for quantitative measure of the identity's strength and lacks, thus, the fulfillment of the sixth law of identity (provision of defining strength of identity). However, the combined approach to provide an umbrella framework for IdM allows Higgins users to choose the best combination of technologies suited to their requirements. Moreover, Higgins architecture is most compliant to the laws of identity (Section II.A) among the state of the art technologies that have been considered in this evaluation. Therefore, in our architecture we have taken some aspects of the Higgins architectural approach and integrated them to our need. In the next section the architecture is illustrated in details.

V. PROPOSED ARCHITECTURE

The Identity Management Module described in this chapter is supposed to be integrated in the Hydra middleware [8] which is a middleware for heterogeneous physical devices in a distributed architecture in ambient environments; the module is named Hydra Identity Manager (HIM).

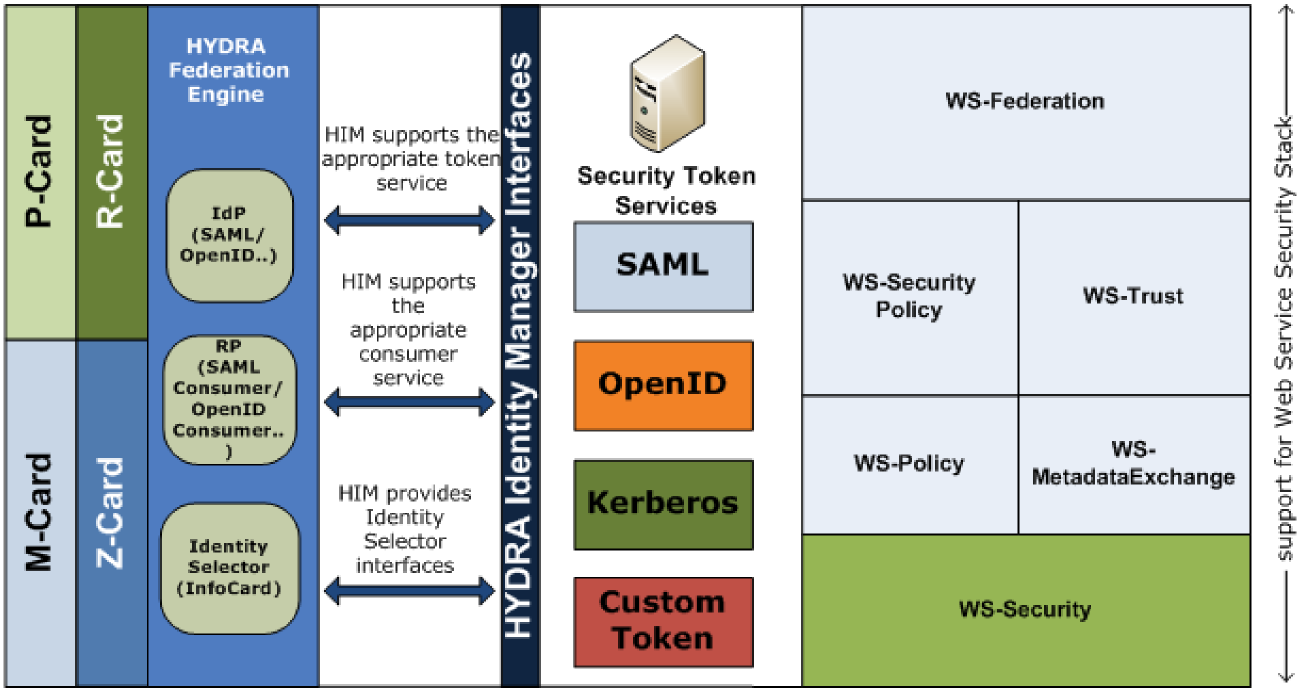

P-Card
R-Card
M-Card
Z-Card
HYDRA Federation Engine
IdP (SAML/ OpenID..)
RP (SAML Consumer/ OpenID Consumer..)
Identity Selector (InfoCard)
HIM supports the appropriate token service
HIM supports the appropriate consumer service
HIM provides Identity Selector interfaces
HYDRA Identity Manager Interfaces
Security Token Services
SAML
OpenID
Kerberos
Custom Token
WS-Federation
WS-Security Policy
WS-Trust
WS-Policy
WS-MetadataExchange
WS-Security
support for Web Service Security Stack

Figure 5: Anatomic view of the proposed architecture of the Hydra Identity Manager.

The architecture of HIM illustrated in our previous paper [16] needed some modifications and improvements due to several reasons. Firstly, the architecture used Windows Communication Foundation (WCF) [5] to take advantage of the out-of-the-box support for WS-Security stack and thus was bound to specific technology. Therefore, the first argument for reengineering the architecture is to redesign it to make it technology agnostic. Secondly, the result of the state of the art evaluation shows that Higgins provides better supports for our law 2 (Minimal Information Disclosure for a Constrained Use) and law 9 (Consistent experience across contexts) with their r-card and z-card concepts. In order to provide the best possible support for the ten laws of identity, there was also the necessity to adopt some further aspects of the Higgins concept into our architecture.

In a service oriented architecture, Hydra's Identity Management System provides support to the developer to implement integrity, confidentiality, and authenticity of such context specific actions, e.g., in work flows, transactions, and processes performed by orchestrated services.

It is important to mention here that the overall architecture of Hydra is designed based on the WS-* family. Because of this technical ground it is necessary for HIM to be compatible to the WS-* family.

Figure 5 shows an anatomic view of the architecture. We propose a hybrid model of existing IdM protocols (SAML, OpenID, InfoCard). This hybrid model enriches the architecture with all round features that are desired by the developer. Moreover, the coexistence of SAML, OpenID, and InfoCard allows to compensate each others' limitations and, thus, to mitigate vulnerabilities. The Hydra Federation Engine supports IdP, RP and Identity Selector of any kind. Moreover, on the client side four different variants of InfoCard are supported. Z-card will bring the user more privacy and r-card will present the user the rich context-aware feature. Since the relationship card will reside on the client machine, the user will have full control over his privacy. Thus, the advantages of intelligent environments and privacy have been put in place in a balanced manner.

The communication viewpoint on the architecture is described in the Figure 6. In this figure the resident of our fictitious scenario is accessing various services in his ubiquitous world. Every node works as IdP and RP thereby realizing the concept of a "Distributed Identity Provider".

"Distributed Identity Provider" means no centralized IdP managing the identity of the user, it is rather the surrounding ubiquitous devices that play the role of IdP and RP back and forth. Authentication to an individual RP has to be realized in 5 basic steps described in the figure. The concept of "Distributed Identity Provider" brings more privacy to the user as no centralized IdP manages the identity of an individual.

In this particular example showed in Figure 6, the subject is being identified by an acceptable IdP (the server) to give access to the laptop. Now once the laptop has the information that the subject has been identified and federated by an acceptable IdP it can take over the

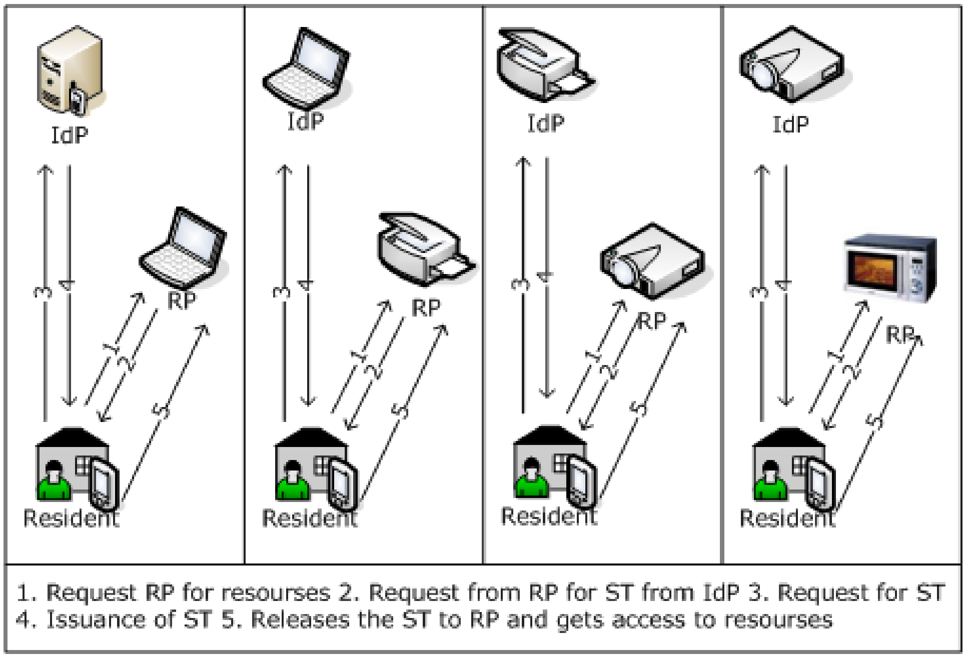

IdP
RP
Resident
3
4
1
2
5
1. Request RP for resourses 2. Request from RP for ST from IdP 3. Request for ST
4. Issuance of ST 5. Releases the ST to RP and gets access to resourses

Figure 6: Communication viewpoint on the architecture of Hydra Identity Manager. The resident of the test scenario (Section II.A) accessing various resources using the principle of identity federation with distributed IdP(s).

role of the IdP and federate the subject further to other RPs e.g., the printer. Similarly, the role of IdP propagates through node to node and each node can act as an IdP. A proper transitive chain is maintained. This example elaborates how "Distributed Identity Provider" can possibly work in a pervasive environment.

A Compliance to the Laws of Identity

The proposed architecture takes an attempt to minimize the vulnerabilities in Identity Management in ambient environments by being compliant to the ten laws of identity we have defined. In this section we will summarize how and up to what extend the architecture fulfills the ten laws.

1) Compliance to the First Law (User Empowerment: Awareness and Control)

The best way invented so far bringing user awareness and control in an Identity Management System is InfoCard. Let it be Windows CardSpace, DigitalMe, SeatBelt or any other InfoCard, when one of these cards pops up before a transaction takes place, it certainly raises the awareness of the user regarding the information he is about to disclose. Moreover, InfoCard enables the user to gain control over his data before disclosure. Thus, it complies with the first law.

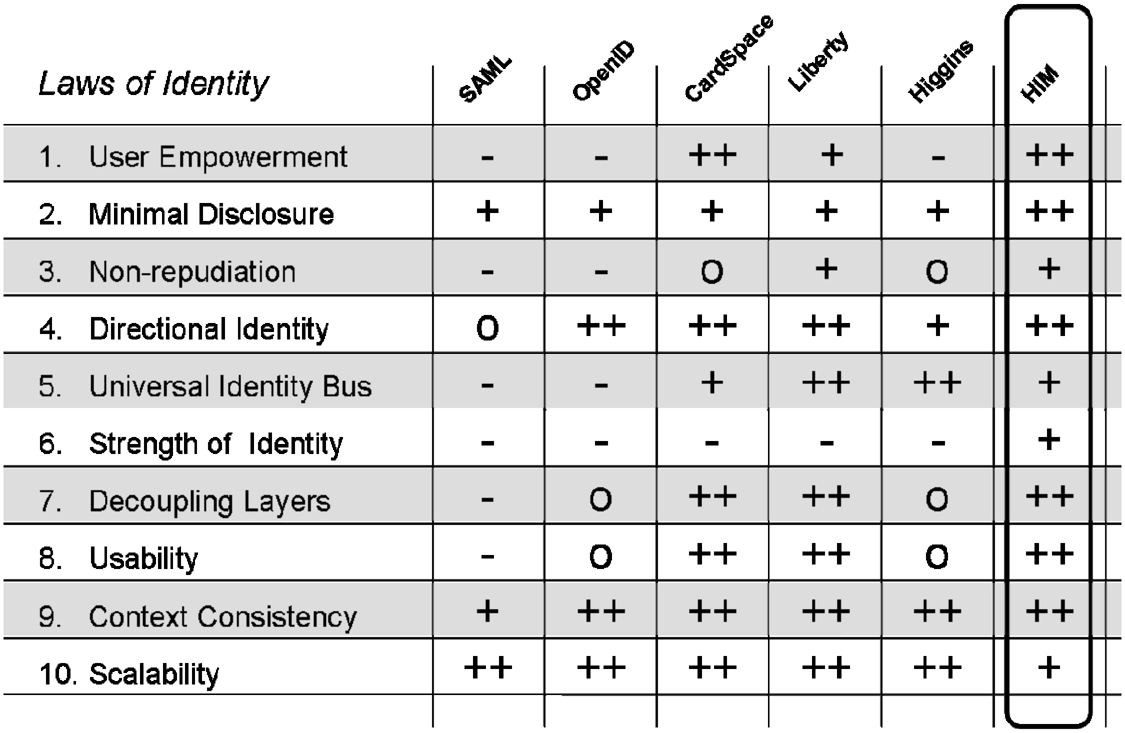
2) Compliance to the Second Law (Minimal Information Disclosure for a Constrained Use)

The STS supports provided in HIM furnishes the developers with SAML, OpenID, Kerberos or even a custom token type. This is how HIM is designed to be token agnostic and virtually supports any token types. With the virtue of SAML and OpenID it is possible to make a bare assertion that a *Subject* has been authenticated by an IdP without having to disclose any information about the *Subject*. This is one feature of HIM that supports the second law.

Another supporting feature for the second law is the zero-knowledge-proof which is an integral part of HIM. The z-card module in the HIM architecture supports the ability of an identity selector to generate zero knowledge proofs that can be conveyed by the agent to the *Relying Party* without revealing any more than it is absolutely necessary and while maintaining the chain of trust back to the original token issuer. As a result, it brings strong support for the second law of identity.

3) Compliance to the Third Law (Non-repudiation)

The three criteria of "non-repudiation" defined in the third law are: authenticity, integrity, and time stamp. WS-Security specification defines all of these three criteria and thus ensures "non-repudiation". Since WS-Security resides at the very bottom of the HIM architecture stack it makes HIM compliant with the third law.



Laws of Identity	SAML	OpenID	CardSpace	Liberty	Higgins	HIM
1. User Empowerment	-	-	++	+	-	++
2. Minimal Disclosure	+	+	+	+	+	++
3. Non-repudiation	-	-	O	+	O	+
4. Directional Identity	O	++	++	++	+	++
5. Universal Identity Bus	-	-	+	++	++	+
6. Strength of Identity	-	-	-	-	-	+
7. Decoupling Layers	-	O	++	++	O	++
8. Usability	-	O	++	++	O	++
9. Context Consistency	+	++	++	++	++	++
10. Scalability	++	++	++	++	++	+

Table 1: Tabular result of the evaluation of state of the art technologies and the proposed architecture.

4) *Compliance to the Fourth Law (Support for directional identity topologies)*

Both SAML 2.0 and OpenID support directional identity and, therefore, law 4 is also satisfied by the proposed hybrid architecture. It is possible to configure SAML 2.0 to implement both bi-directional and unidirectional federation. SAML 2.0 as well as OpenID can be exposed to be omni-directional identity.

5) *Compliance to the Fifth Law (Universal Identity Bus)*

From a developer's viewpoint a UIB is an umbrella platform where he can implement the Identity Management System of his choice and is even able to cross-match different token types, protocols, and information cards. The hybrid architecture of HIM exactly attempts to target such an identity vision for building Identity Management applications.

6) *Compliance to the Sixth Law (Provision of defining strength of identity)*

The sixth law of identity is facilitated by HIM in the "Hydra.IdentityManager.Identity" namespace. Here, depending on the entity and the identity ownership, the relationship of the entity and the strength of the identity is defined.

7) *Compliance to the Seventh Law (Decoupling identity management layer from application layer)*

The Hydra Federation Engine (HFE) acts as the orchestrator of the process of the *Identity Metasystem*. The RP, IdP, and the *Subject* roles are defined in the HFE and, thus, federation is facilitated. This notion of *Identity Metasystem* decouples the Identity Management layer from the rest of the application. HFE in the middleware is what the developers can utilize to achieve federation.

8) *Compliance to the Eighth Law (Usability issue concerning identity selection and disclosure)*

Usability is strongly correlated to the user group and also depends on the nature of the application. Therefore, at a middleware level where the target user group and the nature of the application is not specifically known, it is necessary to facilitate the developer with a wide variety of support to implement usable Identity Management Systems according to his need. HIM gives support for implementing available InfoCards, e.g., CardSpace or DigitalMe. On top of that there is also room for building custom information cards. The developer can then choose the most suited InfoCard in terms of usability.

9) Compliance to the Ninth Law (Consistent experience across contexts)

HIM architecture allows r-card (relationship-card) that manages the users' context experience. These cards can hold different context relevant profiles. An r-card offers a superset of the functionality of an i-card specification by Microsoft. R-cards can be either self-issued, where your identity selector defines and issues the card on your behalf, or issued by a third-party, where an entity other than you defines and issues the r-card. With r-cards, this distinction is less important because in both cases an r-card represents a mutual relationship and agreement to share certain claims/attributes. With the virtue of r-cards the user experiences a context aware smart environment without having to compromise his privacy.

10) Compliance to the Tenth Law (Scalability)

The Hydra Federation Engine is designed in such a way that numerous IdP, RP, and *Subjects* can join in and out and federate identities. At the same time it also supports spawning multiple identities and to manage them in proper ways. This feature enriches the architecture with scalability and, thus, satisfies the tenth law of Hydra identity.

VI. CONCLUSION

The overall comparison of the proposed architecture and the state of the art technologies are presented in Table 1. Since this evaluation is from a middleware viewpoint, it is not justifiable to make a statement that any one of these laws is impossible to realize using one of the existing frameworks. Rather it is more viable to say that some of the frameworks may have strong support to implement one of the laws and on the other hand some of them poorly support that law to be implemented in Identity Management System for ambient environment. That is why we came up with a scale of poor (-) to very good (++) and stated the result in Table 1.

In this paper we have presented an architectural approach to tackle the challenges of Identity Management in ubiquitous computing. The hybrid architecture presented has been adopted from the existing standardize state of the art IdM technologies. The future plan is to implement the architecture and integrate it in the Hydra middleware. The Higgins framework has been chosen for implementation based on the result of the evaluation.

REFERENCES

[1] Cameron, K, Laws of Identity (2005), Microsoft Corporation, last access May 2009.

[2] Mercuri, M. 2007 *Beginning Windows Cardspace: from Novice to Professional*. Apress.

[3] Recordon, D. and Reed, D. 2006. "OpenID 2.0: a platform for user-centric identity management", in *Proceedings of the Second ACM Workshop on Digital Identity Management* (Alexandria, Virginia, USA, November 03 - 03, 2006). DIM '06. ACM, New York, NY, 11-16. DOI= http://doi.acm.org/10.1145/1179529.1179532

[4] Rosenberg, J. and Remy, D. 2004 *Securing Web Services with Ws-Security: Demystifying Ws-Security, Ws-Policy, Saml, XML Signature, and XML Encryption*. Pearson Higher Education.

[5] McMurtry, C., Mercuri, M., Watling, N., and Winkler, M. 2007 *Windows Communication Foundation Unleashed (Wcf) (Unleashed)*. Sams.

[6] OpenID, http://openid.net/

[7] Hydra, Deliverable D2.1a Scenarios for usage of Hydra in Building Automation, 25 January 2007 - version 1.41.

[8] The Hydra Project, http://www.hydramiddleware.eu

[9] Vedamuthu, A. S., Orchard, D., Hondo, M., Boubez, T., Yendluri, P., Web Services Policy 1.5 – Primer, W3C Working Draft 18 October 2006, http://www.w3.org/TR/2006/WD-ws-policy-primer-20061018

[10] Artman, H. 2002. Procurer usability requirements: negotiations in contract development. In *Proceedings of the Second Nordic Conference on Human-Computer interaction* (Aarhus, Denmark, October 19 - 23, 2002). NordiCHI '02, vol. 31. ACM, New York, NY, 61-70. DOI= http://doi.acm.org/10.1145/572020.572029

[11] *WS-Trust 1.3*, OASIS Standard 19 March 2007, http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.html#_Toc162064937

[12] Akram, H., Hoffmann, M., *Laws of Identity in Ambient Environments: The Hydra Approach*, The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies UBICOMM 2008, September 29 - October 4, Valencia, Spain

[13] Oh, Hyun-Kyung; Jin, Seung-Hun, "The Security Limitations of SSO in OpenID," *Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference on* , vol.3, no., pp.1608-1611, 17-20 Feb. 2008

[14] Alrodhan, W.A.; Mitchell, C.J., "Addressing privacy issues in CardSpace," *Information Assurance and Security, 2007. IAS 2007. Third International Symposium on* , vol., no., pp.285-291, 29-31 Aug. 2007

[15] Anderson, R. J. 2008 *Security Engineering: a Guide to Building Dependable Distributed Systems*. 2nd. John Wiley & Sons, Inc.

[16] Akram, H., Hoffmann, M., *Supports for Identity Management in Ambient Environments: The Hydra Approach*, International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services I-CENTRIC 2008 October 26-31, 2008 - Sliema, Malta

[17] Maler. E., SAML, Liberty Alliance, openLiberty, and Concordia, Sun Microsystems, Inc, 2007.

[18] *On the Insecurity of Microsoft's Identity Metasystem CardSpace*, Press release, Bochum, Germany, May 27, 2008, http://demo.nds.rub.de/cardspace/PR-HGI-TR-2008-003-EN.pdf

[19] McCullagh, A., Caelli, W., *Non-Repudiation in the Digital Environment*, First Monday, volume 5, number 8 (August 2000), URL:

http://firstmonday.org/issues/issue5_8/mccullagh/index.html

[20] *Web Services Metadata Exchange (WS-MetadataExchange), Version 1.1*, August 2006, http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-mex/metadataexchange.pdf

[21] *Web Services Federation Language (WS-Federation),* Version 1.1, December 2006, http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-fed/WS-Federation-V1-1B.pdf?S_TACT=105AGX04&S_CMP=LP

[22] Hodges, J., *Technical Comparison: OpenID and SAML - Draft* 06, January 17, 2008, http://identitymeme.org/doc/draft-hodges-saml-openidcompare-06.html#tbl-exec-summary

[23] Armando, A., Carbone, R., Compagna, L., Cuellar, J., and Tobarra, L. 2008. Formal analysis of SAML 2.0 web browser single sign-on: breaking the SAML-based single sign-on for google apps. In *Proceedings of the 6th ACM Workshop on Formal Methods in Security Engineering* (Alexandria, Virginia, USA, October 27 - 27, 2008). FMSE '08. ACM, New York, NY, 1-10. DOI= http://doi.acm.org/10.1145/1456396.1456397

[24] E. Maler, SAML basics - A technical introduction to the Security Assertion Markup Language, http://www.itu.int/itudoc/itu-t/com17/tutorial/85573.html

[25] Eve Maler et al. Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1 Oasis-Open, September 2003. OASIS Standard, http://www.oasisopen.org/committees/security/

Intelligent Electronic Nose Systems with Metal Oxide Gas Sensors for Fire Detection

Michifumi Yoshioka, Toru Fujinaka, Sigeru Omatu
Graduate School of Engineering, Osaka Prefecture University
Naka-ku, Sakai, 599-8531, Osaka, Japan
yoshioka, fuji, omatu@cs.osakafu-u.ac.jp

***Abstract*— In this paper, a reliable electronic nose system designed from the combination of various semi-conductor metal oxide gas sensors (MOGS) is applied to the detection of fire resulting from various sources in a kitchen. The time series signals obtained from the same source of fire are highly correlated, and different sources of fire exhibit unique patterns in the time series data. Therefore, the error back-propagation (BP) method can be effectively used for the classification of the tested smell. The accuracy of 99.6% is achieved by using only a single training data set from each source of fire. The accuracy achieved with the *k*-means algorithm is 98.3%, which also shows the high ability of the EN in detecting the early stage of fire from various sources.**

***Index Terms*— electronic nose, neural networks, learning vector quantization, metal oxide gas sensor, smell classification**

I. Introduction

Over the last decade, odor-sensing systems (so-called electronic nose systems) have undergone important developments from technical and commercial viewpoints. The electronic nose (EN) refers to a device of reproducing human sense of smell based on sensor arrays of smell and pattern recognition methods. Recently, there are several commercial EN instruments currently in use in the world such as quality control of food industry [1], environmental protection [2], public safety [3] and space applications [4].

Every year the damage from the household fire disaster brings about not only severe loss to property assets, but also physical and psychological injuries of the people.

In this paper, we will explain the human olfactory process and the EN system. After surveying various types of the odor sensors, we explain the and then the reliability of a new EN system developed from various semi-conductor metal oxide gas sensors (MOGSs) is presented to specify the smell from various sources of fire.

James A. Milke [5] has proved that two kinds of MOGS have the ability to classify several sources of fire more precisely compared with conventional smoke detector. However, his results achieve only 85% of correct classification.

In this paper, a new EN [6] is applied to measure smells from various sources of fire such as household burning materials, cooking smells, the leakage from the liquid petroleum gas (LPG). The new EN has been successfully applied to the classification of not only similar smells from different kinds, but also the same kind of smell at different concentration levels. The time series signals of the MOGS from the beginning to the time until the MOGS fully adsorbs the smell from each source of fire are recorded and analyzed by the error back-propagation (BP) neural network and the *k*-means algorithm. The average classification rate of 99.6% can be achieved by the BP method with only a single training data set from each source of fire. The accuracy with *k*-means algorithm is 98.3%, which is much better than the results in [5]– [9]. These results confirm the reliability of this new device in detecting various sources of fire in the early stage.

II. Human Olfactory Processes

Although the human olfactory system is not fully understood by physician, the main components about the anatomy of human olfactory system are the olfactory epithelium, the olfactory bulb, the olfactory cortex, and the higher brain or cerebral cortex.

The first process of human olfactory system is to breathe or to sniff the smell into the nose. The difference between the normal breath and the sniffing is the quantity of odorous molecules that flows into the upper part of the nose. In case of sniffing, most air is flown through the nose to the lung and about 20% of air is flown to the upper part of the nose and detected by the olfactory receptors.

In case of sniffing, the most air is flown directly to the upper part of the nose and interacts with the olfactory receptors. The odorous molecules are dissolved at the mucous layer before interacting with olfactory receptors in the olfactory epithelium. The concentration of odorous molecules must be over the recognition threshold. After that, the chemical reaction in each olfactory receptor produces an electrical stimulus. The electrical signals from all olfactory receptors are transported to olfactory bulb. The input data from olfactory receptors are transformed to be the olfactory information to the olfactory cortex. Then the olfactory cortex distributes the information to other parts to the brain and human can recognize odors precisely. The other parts of the brain that link to the olfactory cortex will control the reaction of the other organ against the reaction of that smell. When human detects bad smells, human will suddenly expel those smells from the nose and try to avoid breathing them directly without any protection. This is a part of the reaction from the higher brain. The next process is to clean the nose by breathing fresh air to dilute the odorous molecules until those concentrations are lower than the detecting threshold. The time to dilute the smell depends on the persistence qualification of the tested smell.

The processes to analyze smell by a human nose can be summarized by a diagram shown in Fig. 1.

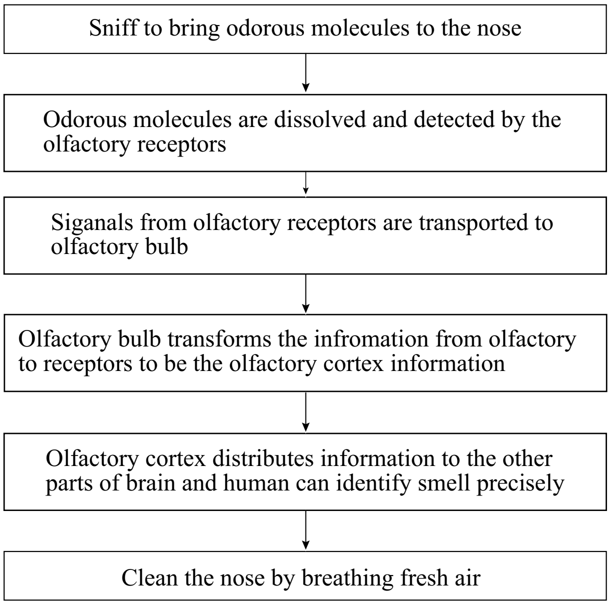

Sniff to bring odorous molecules to the nose
Odorous molecules are dissolved and detected by the olfactory receptors
Siganals from olfactory receptors are transported to olfactory bulb
Olfactory bulb transforms the infromation from olfactory to receptors to be the olfactory cortex information
Olfactory cortex distributes information to the other parts of brain and human can identify smell precisely
Clean the nose by breathing fresh air

Fig. 1. Olfactory system

III. EN SYSTEM

The EN systems provide an alternative method to analyze smell by imitating the human olfactory system. In this section, the concept of an EN is explained. Then various odors sensors applied as the olfactory receptors are explained. Finally, the mechanism of a simple EN that was developed in this paper is described in detail by comparing the function of each part with the human olfactory process.

A. EN concept

The mechanism of EN systems can be divided into main four parts as shown in Fig. 2.

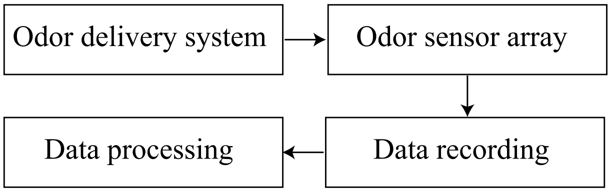


Fig. 2. Main parts of EN systems.

1) Odor delivery system: The first process of human olfactory system is to sniff the odorous molecule into the nose. Thus, the first part of the EN system is the mechanism to bring the odorous molecules into the EN system. There are three main methods to deliver the odor to the EN unit, sample flow, static system, and preconcentration system.

The sample flow system is the most popular method to deliver odorous molecule to the EN unit. Some carrier gas such as air, oxygen, nitrogen, and so on, is provided as a carrier gas at the inlet port to flow the vapor of the tested smell through the EN unit via the outlet port. The mechanism to control the air flow of an EN may contain various different parts such as a mass flow controller to control the pressure of the carrier gas, a solenoid valve to control the flow of inlet and outlet ports, a pump to suck the tested odor from the sampling bag in case that the tested odor is provided from outside, a mechanism to control humidity, and so on. Most commercial ENs contain complicated odor delivering systems and this makes the price of the ENs become expensive.

The static system is the easiest way to deliver odorous molecules to the EN unit. The EN unit is put into a closed loop container. Then an odor sample is injected directly to the container by a syringe. It is also possible to design an automatic injection system. However, the rate to inject the test odors must be controlled to obtain accuracy results. Normally, this method is applied for the calibration process of the EN. But in this case the quantity of the odor may not be enough to make the sensor reach the saturation stage, that is, the stage that sensor adsorbs the smell fully.

The preconcentration system is used in case of the tested smell that has a low concentration and it is necessary to accumulate the vapor of the tested odor before being delivered to the EN unit. The preconcentrator must contain some adsorbent material such as silica and tested odor is continuously accumulated into the preconcentrator for specific time units. Then the preconcentrator is heated to desorb the odorous molecule from the adsorbed material. The carrier gas is flown through the preconcentrator to bring the desorbed odorous molecules to the EN unit. By using this method, some weak smells can be detected by the sensor array in the EN unit.

2) EN unit: All ENs must contain an array of odor sensors which act like the human olfactory receptors. Various kinds of odor sensors can be used to detect odors. The details of odor sensors will be explained in the following section. The number of sensors in the ENs unit is around 4 to 50 sensors depending on the design of each EN. The electrical circuit to control the input and the output of the sensor array are varied on the types of sensors. It is also possible to use several sensors from the same model for analyzing the odor just like the human olfactory receptors that may contain the same kind of receptors from different ages (human olfactory replacement time is around 30 days as an average). Inside of the EN units may contain some mechanism to clean the sensor after finishing the testing process in order to speed up the EN to analyze new smell.

3) Data recording: It is necessary to record the output of the sensor array in the EN unit while absorbing the tested odor continuously. Generally, the output from the sensor array is analog signal and it is necessary to convert the analog signal to digital signal before being recorded in the computer memory starage and A/D converters are used which include the software to control the sampling time with the user interface. This part of the EN system is like the function of olfactory bulb to collect the signal from the olfactory receptors before passing to the olfactory cortex.

4) Data processing: Once the data from the tested odor are completely available for analyzing, it may be able to apply various methods to analyze these data such as artificial neural networks(NNs)or statistical analysis. It may be necessary to modify the]raw data from the sensor array to increase the speed of data processing, that is, training time for the supervised NNs, or to correct some variation of data due to the effect of measuring environment. The way to modify the data is just like the internal function of olfactory bulb before passing the input to the olfactory cortex. The NNs or the statistical analysis with the software to interface with the user will act like the olfactory cortex and cerebral cortex in the human olfactory system.

The data processing part is one of the most important step to make the EN system become reliable. Most commercial ENs have a complicated mechanism for the odor delivery system and the EN unit, but if the data processing part of those ENs are not well designed, the high reliability of the ENs system may not be achieved. This paper will focus on the data processing to classify the several smells precisely.

IV. Odor Sensors

Actually there are many kinds of sensors that can be used to detect odorous molecules. However, only a few kinds of them have been successfully applied as artificial olfactory receptors in commercial ENs. Those sensors are a conducting polymer (CP)[1], a quartz crystal microbalance (QCM)[6], a surface acoustic wave (SAW)[1], and MOGSs[6].

A. Conducting polymer (CP) sensor

The CP sensor is widely applied as the array of artificial olfactory receptor in the commercial EN. These sensors are made by a thin film electro-polymerization of the conducting polymer across the gap between two inert electrodes. Variety of different monomers can be polymerized to give conducting polymer film.

CPs show reversible changes in conductivity when chemical substances such as methanol, ethanol, and ethyl acetate adsorb and desorb from the polymer. However, the mechanism that the conductivity is changed by this adsorption is not clearly understood like the case of MOGSs. Anyway, variety of polymers film responds to different kinds of odor. The choices of CPs sensors are wider than those of MOGSs. CP sensor is operated at the nearly room temperature and it consumes low power consumption. However, the response to odorous molecule is slower than those of MOGSs and it is really sensitive to humidity. The CP sensors are used in some commercial ENs, but only the sensor itself is not sold separately.

B. Quartz crystal microbalance (QCM) sensor

The QCM sensor is a kind of acoustic wave gas sensors. The main element of a QCM sensor is made from a slice of single crystal quartz typically around 1 cm in diameter. The electrodes, usually gold, are coated on both surfaces of the crystal quartz. The suitable alternative voltage is applied across the two electrodes. The electromechanical effect can make the crystal oscillate at its resonant frequency.

The oscillation wavelength and resonant frequency are altered depending on the thickness (weight) of the crystal quartz. The acoustic waves are oscillated at ultrasonic frequency typically around 1 to 500 MHz. The sensitivity of the QCM sensor to the odorous molecules depends on the gas sensitive coating material on the surface of the crystal quartz. The adsorbent coating materials are generally electro-polymers since the wide ranges of materials must be synthesized. When the adsorbent coating material adsorbs odorous molecules, the wave velocity, frequency, and the amplitude of oscillation of the sensor are changed. Therefore, the QCM with various kinds of coating materials can be used as an array of artificial olfactory receptors in the EN.

C. Surface acoustic wave (SAW) sensor

A surface acoustic wave (SAW) sensor is another kind of acoustic wave sensor that is similar to the QCM sensor. The SAW sensor comprises with a thick plate of crystal quartz. A surface of the SAW is coated with the gold electrodes to excite the oscillation of the surface of sensor. The wave is generated by applying an alternative voltage to the gold electrodes on the surface of the sensors. There are two electrode pairs, one is for transmitting the wave and the other pair is for receiving the acoustic wave.

The adsorbent coating material is coated on the surface between the transmitter and receiver electrode pairs. The changes in frequency of acoustic wave on the surface of the SAW sensors can be detected while the adsorbent material adsorbing odorous molecules. The properties of acoustic are affected by the changes in the properties of the crystal surface. The operating frequency of an SAW is between 30 to 300 MHz. The sensor also requires an integrated circuit to generate and receive acoustic wave like the QCM sensor. SAW sensor is sensitive to humidity and the response time that is not as fast as the MOGS and the CP sensor.

D. Semi-conductor metal oxide gas sensors (MFGOGSs)

MOGS is the most widely used sensor for making an array of artificial olfactory receptors in the EN system. These sensors are commercially available as the chemical sensor for detecting some specific smells. Generally, an MOGS is applied in many kinds of electrical appliances such as a microwave oven to detect the food burning, an alcohol breath checker to check the drunkenness, an air purifier to check the air quality, and so on.

Therefore, we will use MOGSs as the smell classification in what follows. The picture of some commercial MOGSs are shown in Fig. 3.

Various kinds of metal oxide, such as SnO_2, ZnO_2, WO_2,TiO_2 are coated on the surface of semi-conductor, but the most widely applied metal oxide is SnO_2. These metal oxides have a chemical reaction with the oxygen in the air and the chemical reaction changes when the adsorbing gas is detected. The scheme of chemical reaction of an MOGS when adsorbing with the CO gas is shown as follows:

$$\frac{1}{2}O_2 + (SnO_{2-x}) \rightarrow O^- ad(SnO_{2-x}) \tag{1}$$

$$CO + O^- ad(SnO_{2-x}) \rightarrow CO_2 + (SnO_{2-x}) \tag{2}$$

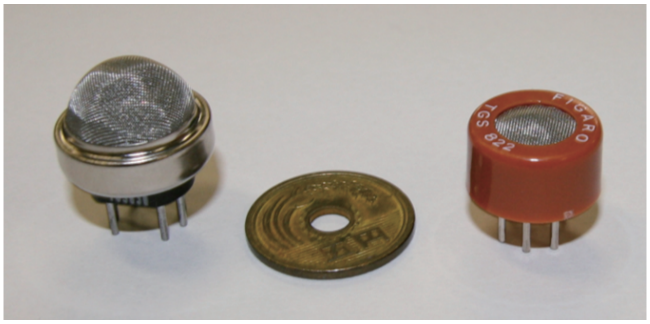


Fig. 3. MOGS

When the metal oxide element on the surface of the sensor is heated at a certain high temperature, the oxygen is adsorbed on the crystal surface with the negative charge as shown in Fig. 3. In this stage the grain boundary area of the metal oxide element forms a high barrier as shown in the left hand side of Fig. 3. Then the electrons cannot flow over the boundary and this makes the resistance of the sensor become higher. When the deoxidizing gas, e.g., CO gas, is presented to the sensor, there is a chemical reaction between negative charge of oxygen at the surface of the metal oxide element and the deoxidizing gas as shown in (2). The chemical reaction between adsorbing gas and the negative charge of the oxygen on the surface of MOGS reduces the grain boundary barrier of the metal oxide element as shown in the right hand side of 4. Thus, the electron can flow from one cell to another cell easier. This makes the resistance of MOGS lower by the change of oxygen pressure according to the rule of (3).

The relationship between sensor resistance and the concentration of deoxidizing gas can be expressed by the following equation over a certain range of gas concentration:

$$R_s = A[C]^{-\alpha} \tag{3}$$

where R_s =electrical resistance of the sensor, A = constant, C =gas concentration, and α =slope of R_s curve.

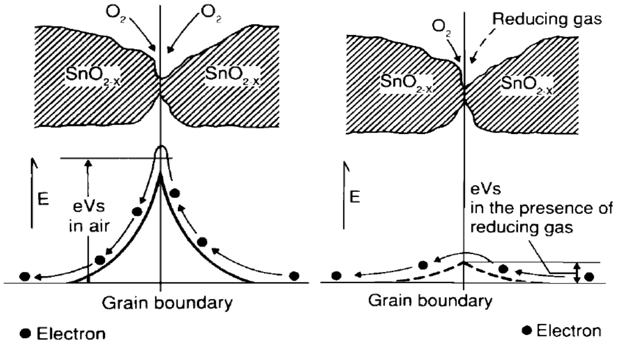


Fig. 4. Principle of MOGS[7]

The electric circuit for the MOGS is shown in Fig.5. Electrical voltages are provided to the circuit(V_c) and the heater of the sensor(V_h). When the MOGS is adsorbed with oxygen and the deoxidizing gas, the resistance of the sensor (R_s) is changed. Thus, it can measure the voltage changes while the sensor adsorbing the tested odor(V_{out}).

MOGSs need to be operated at high temperature, so they consume a little higher power supply than the other kinds of sensors. The reliability and the sensitivity of MOGSs are proved to be good to detect volatile organic compounds (VOCs), combustible gas, and so on [7]. However, the choices of MOGSs are still not cover all odorous compounds and it is difficult to create an MOGS that responds to one odor precisely. Generally, the commercial MOGS responds to various odors in different ways. Therefore, we can expect if we use many MOGSs to measure a smell, the vector data reflect the specific properties for the smell.

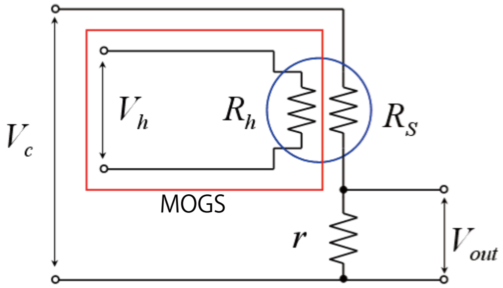


Fig. 5. Principle of MOGS[7]

Generally, it is designed to detect some specific smell in electrical appliances such as an air purifier, a breath alcohol checker, and so on. Each type of MOGS has its own characteristics in the response to different gases. When combining many MOGS together, the ability to detect the smell is increased. An EN system shown in Fig. 6 has been developed, based on the concept of human olfactory system stated above. The combination of MOGS, listed in Table I, are used as the olfactory receptors in the human nose. The MOGS unit is combined with the air flow system to lead the air and the tested smell into the MOGS unit. The A/D converter transforms the analog signals to digital signals and stores them in the data recording system before being analyzed by multivariate analysis methods, such as the BP method and the *k*-means algorithm.

TABLE I
LIST OF MOGS FROM THE FIS INC. USED IN THE EXPERIMENT

Sensor Model	Main Detecting Gas
SP-53	Ammonia, Ethanol
SP-MW0	Alcohol, Hydrogen
SP-32	Alcohol
SP-42A	Freon
SP-31	Hydrocarbon
SP-19	Hydrogen
SP-11	Methane, Hydrocarbon
SP-MW1	Cooking vapor

The main part of the MOGS is the metal oxide element on the surface of the sensor. When this element is heated at a certain high temperature, the oxygen is adsorbed on the crystal surface with the negative charge. The reaction between the

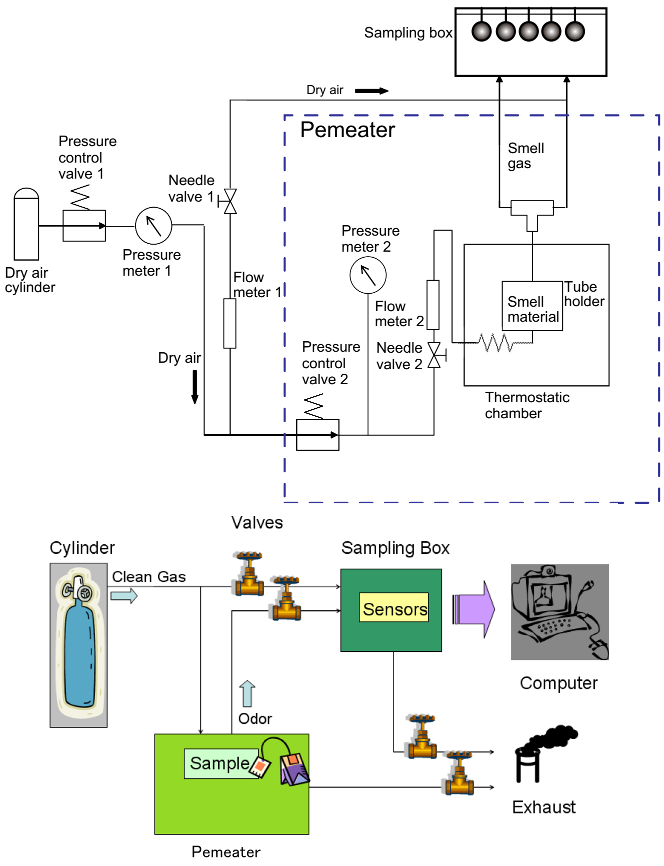


Fig. 6. Structure of the electronic nose system

negative charge of the metal oxide surface and deoxidizing gas makes the resistance of the sensor vary as the partial pressure of oxygen changes[7]. Based on this characteristic, we can measure the net voltage changes while the sensors adsorb the tested odor.

V. Experimental Data Collection

The investigation of the EN under temperature and humidity controlled environments is performed and the experiment is done in winter season.

1) Experimental conditions : Each data is repeatedly tested consecutively forty times under similar weather conditions to keep the temperature and the humidity constant. The concentration of all tested smells is controlled as constant as possible in order to observe the repeatability qualification of the MOGSs in this EN.

The temperature and humidity of normal air during the test period are controlled by adjusting the valves shown in Fig. 6. The carrier gas in this experiment is the dry air. Since the measuring environment is controlled by using the permeator, which can control the temperature, flow rate of the air, and humidity, we assume that the measurement data do not depend on the environment in this experiment.

2) Measuring method : There are two methods to measure the smell. The first method called “ transient method (TM)” is done by injecting the tested smell into the EN for a short period of time, for example, 120s, while measuring the smell. The other method called “saturation method (SM)” is done by continuously providing the tested smell to the EN until the smell data approach the saturation stage.

3) Measuring data : The smell from twelve sources of fire listed in Table II are measured by the EN system explained in the previous section. Each source of fire has been tested with forty repetition of data measured on different days, in order to check the repeatability in the response of the MOGS to the same smell.

TABLE II
List of Burning Materials in the Experiment

Sources of fire	Abbreviation
Steam from boiling water	Steam
Burning joss stick	Joss
Burning mosquito coil	Mos
Aroma oil	Aroma
Aroma candle	Candle
Flame from liquid petroleum gas(LPG)	Flame
Leakage of LPG	LPG
Steam from Japanese soup “oden”	Oden
Boiling vegetable oil	Oil
Toasted bread	Toast
Burning paper	Paper
Burning wood	Wood

For each data set, the voltage signal in the dry air is measured every two seconds for a sensor s during one minute, and its average value $\bar{V}_s(air)$ is used as the reference point. After that, a testing smell is adsorbed, and the voltage $V(s,t)$ is measured every two seconds for a period of two minutes on each smell sample. Finally, the net change of the voltage signal in each period, $V_{smell,t}$, is calculated by

$$V_s(t) = V(s,t) - \bar{V}_s(air), s = 1, 2, ..., 8 \tag{4}$$

where t is the time from 0 to 120s and s denotes a kind of sensor.

After testing one smell the MOGS need to be cleaned by removing the tested smell and supplying only the fresh air until the output of the MOGS return to a stable point and a new sample can be tested. This process is just like the human nose which needs to breath the fresh dry air before being able to recognize a new smell accurately. Some time series data are shown in Fig. 7. From these data, we can see that many time series data of smells approach the saturation stages within the measuring periods. But Some smells can not approach the saturation stages within the measuring period, such as OIL, PAPER, WOOD, because of two reasons. First reason is that the distance from the tested smell to the EN unit is increased and it takes longer time for smell to flow to the EN unit. The second reason is the changes of smell of the burning material. For example, the smell of OIL is stronger and stronger if heating is provided to the container of the vegetable oil. The vegetable oil is able to turn to be the flame of fire if the heating is continuously provided to the container of the tested oil. Thus, we decided to measure the smell only two minutes even though the data signals of some smells can not reach the saturation stage.

The smells from AROMA, CANDLE, and the FLAME are not as strong as the other tested smell. Thus, the responses of the data signals from these data are lower than the other smells. The signals from the same smell in every repetition are similar and each smell has a unique pattern. However, some smells like PAPER, OIL, and WOOD have some repetition data of different patterns from their main repitition data due to the inconsistent burning rate of these smells. Since all tested smells in every repetition data are measured under similar environment, it is not necessary to apply the normalization method to modify the data. In this case, the raw data obtained from Eq.(4) can be used as the input data for the NNs or the statistical analysis, directly.

The method to measure the correlation of the data will be performed in order to choose the proper training data for the NN with the BP algorithm. Furthermore, the signals from the same source of fire in every repetition data are similar in most data sources. The results using the BP method and the *k*-means algorithm to analyze the time series data from each source of fire every two seconds and the average signals during the saturation stages (time 100–120s) are discussed.

VI. Correlation of the Experimental Data

Before classifying each source of data, the correlation of each data source is investigated by using the similarity index (SI) and the principal components analysis (PCA).

A. Similarity Index

In the statistical application, the correlation value developed mainly by Karl Pearson is widely used to find the relationship between two random variables. In this paper, we call the correlation value as a similarity index (SI). The SI value varies from -1 to $+1$. Two random variables with an SI of either -1 or $+1$ are highly correlated, because the knowledge of one provides precise knowledge of the other. However, the SI provides information only for linear relationships between random variables. Random variables could have a nonlinear relationship but still have a SI close to zero [4]. Therefore, we make an assumption on this application that each data pattern has nearly linear relationship to the other data patterns. The SI value between two data is calculated by

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \qquad (5)$$

where r_{xy} is the the SI value, x and y are the comparing data, $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$, $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$, and n is the the size of each data set.

In this experiment, sensor SP-MW1 has some responses with the smell that has high humidity such as the STEAM, ODEN, and LPG. Thus, the signal from this sensor is included as the analytical data. Therefore, the dimension of data vector, n, in this experiment is equal to 480 (8 sensors×60 samples).

The SI values of the data that have consistent burning rates such as the JOSS and the MC are higher than 0.995 in every compared data. The data with inconsistent burning rates like the PAPER and the WOOD have lower SI values in some pairs of data. This means that the repetitions of these data types are less correlated than the data with high SI values. Later we will show that the classification rate of the data with high SI values is higher than those of the data with low SI values.

Since a pair of data with high SI values has linear relationship to each other, the signals from the same smell in every repetition data have high SI values. It means that the EN has reproducibility qualification under similar measuring environment. Thus, it may be able to use only a single data from each smell to be training data of the NN by the BP method or the initial data vector for the statistical analysis like the *k*-means algorithm. Only a single data from each smell with the highest average SI value to the other repetition data of its own smell is selected as the training data.

B. Principal Components Analysis (PCA)

The PCA is applied to transform the time series data by using the input data obtained from Eq.(4). The dimension of the input data is equal to four hundred and eighty. The plot of two main components from the PCA is shown in Fig. 8 where two cases of the experimental data are shown. The loading factors of principal components 1, 2 are about 64%,16% and about 71%, 19%, for full time series stage and saturation stage, respectively. The full time series stage data use the data signals every two seconds, and the saturation stage data case uses only the average data from time 100s to 120s for the analysis. The distributions of the data with inconsistent burning rate such as the PAPER and the WOOD are more scattered than the other data with consistent burning rates such as the JOSS, especially in the saturation stage data. Most of the tested data are separated into their own clusters with some overlap zones among different data sources.

C. k*-means Algorithm*

The *k*-means algorithm is a kind of statistical method. It is an unsupervised learning method that assign an input vector to the nearest cluster center based on the Euclidean distance. The *k*-means algorithm is performed as follows:

Step 1. Select an initial set of cluster centers, $C_i = (c_{i1}, c_{i2}, \cdots, c_{ij})$ where i and j are an index of cluster and a dimension of the input vector. The number of clusters in the input vector space is assumed to be known such as k. The cluster center $C_i, i = 1, 2, ..., k$ are selected from a sample of the input vector that should belong to that cluster.

Step 2. Assign an input vector l of $X_l = (x_{l1}, x_{l2}, \cdots, x_{lj})$ to its closest cluster index by calculating the Euclidean distance between the input vector X_l and each cluster center, $C_i, i = 1, \cdots, k$. Here, x_{lm} denotes an input data from cluster index l and *m*-th component of the input vector.

Step 3. Repeat *Step 2* until all n input vectors are assigned to any cluster.

Step 4. Compute the new cluster centers by calculating the mean value of all input vectors that belong to that cluster.

Step 5. If the position of any cluster center changes, return to *Step 2.* Otherwise stop.

The processing time of the *k*-means algorithm is much faster than the training time of the NN by the BP method. Thus, it is a useful method to analyze the data that have high dimension size like this experiment.

VII. Classification Results Using Full Time Series Data

Full time series data is obtained from Eq.(4) with the dimension of four hundred and eighty that is equal to the number of sensors times the measurement data (8×60). They are used as the input data for the NN by the BP method as well as the data of *k*-means algorithm.

A. Experimental Results Using the NN

The structure of the NN by the BP method as shown in Fig. 9 has three layers. The input layer contains four hundred and eighty input neurons that are equal to the number of sensors (8) tomes the sampling number (60) where the sampling time is 2[s] and the measurement interval is 2[min]. The hidden layer has been tried with several values and finally forty hidden nodes give the best result. The output layer contains twelve nodes where each node represents the smell from each source of fire as listed in Table II. The learning rate, α, and the momentum rate, μ, during the training period are set by trail and error to o.1 and 0.001, respectively.

After checking the correlation of the data by applying the SI value and the PCA, it is found that most data are highly correlated to its own repetition data and most data are distributed densely in its own area with some overlap zone as shown in Fig. 8. The smells that have consistent burning rate like JOSS, MC, LPG have the average SI value between their repetition data higher than 0.995. This means that these data have highly linear relationship with their own repetition data.

In the other way, some smells that have inconsistent burning rate like PAPER or WOOD have some scattered repetition data and these data have the average SI value to their repetition data only around 0.980 which is not as high as the average SI value of JOSS or MC. However, this experiment decides to choose only a single data from each smell in Table II to use as the training data for the NN by the BP method. This data is selected by choosing the data that has the highest average SI value to its own repetition data. The training process is started by the randomized initial weight, and the process to update the weights continues until the minimum mean squared error is less than or equal to 0.0003.

In this experiment, it is assumed that a smell is classified correctly if the highest value of output node is greater than or equal to 0.6 while the corresponding reference value is equal to 1, and those of other output nodes are equal to 0. The results obtained by NN with the BP algorithm are shown in Table IV.

B. Experimental Results Using the k-Means Method

The *k*-means method is applied to analyze the same data as in the case of NN with the BP algorithm. At first, the number of cluster is set to be twelve. Each cluster represents a tested smell shown in Table II. The initial set of cluster center uses a data vector from each smell by choosing the data that has the highest SI value to its own repetition data. All input data have been assigned to each cluster and the processes to determine the new cluster center are performed by the *k*-means algorithm stated above. The final classification results using the *k*-means method are shown in Table IV

TABLE III
RESULTS USING NN

Sources of fire	Correct rate	Correct%	Output value
STEAM	39/39	100	0.988
JOSS	39/39	100	0.997
MOS	39/39	100	0.995
AROMA	39/39	100	0.995
CANDLE	39/39	100	0.963
FLAME	38/39	100	0.978
LPG	39/39	100	0.998
ODEN	39/39	100	0.992
OIL	34/39	100	0.987
TOAST	39/39	100	0.982
PAPER	37/39	94.9	0.935
WOOD	39/39	100	0.964
Average		**99.6**	**0.982**

TABLE IV
RESULTS USING THE *k*-MEANS METHOD

Sources of fire	Correct rate	Correct%
STEAM	40/40	100
JOSS	40/40	100
MOS	40/40	100
AROMA	40/40	100
CANDLE	40/40	100
FLAME	40/40	100
LPG	40/40	100
ODEN	40/40	100
OIL	40/40	100
TOAST	40/40	100
PAPER	35/40	87.5
WOOD	37/40	92.5
Average		**98.3**

C. Discussions of both results by NN and the k-means methods

The results from the NN method have only two incorrectly classified data from paper burning smell (PAPER). These data are not misclassified as the other smells, but only the value of their output nodes are not high enough to classify them as the paper burning smell.

VIII. Another Experiments

The input data are changed in the two cases:1) number of sensors are 4 and 4 to check the effect of classification according to the number of input data(Experiment 1), 2) number of sensors are eight and the input data has been characterized by the regression coefficients (Experiment 2).

A. Result in case of four and six sensors

We have used four and six sensors for this experiment. The former is SP-MW1, SP-31, SP-32, and SP-42A in Table 1 and the latter is SP-MW1, SP-19, SP-31, SP-32, SP-53, and SP-42A in Table 1. In the former case total number of the input data is 240 samples for 4 sensors since the sampling time is 2 [s] and the measurement time is 2 [min]. In the latter case it has 360 samples for 6 sensors. For the hidden layer, we have tried several values, and the number of nodes which gives good accuracy and reasonable training time for both data turned out to be forty. The output layer contains twelve nodes, each node representing one data source. The learning rate, the momentum rate, and the minimum mean square error (MSE) during the training period are set by trial and error method to 0.1, 0.001, and 0.0003, respectively.

Two case of data are analyzed by the BP method and *k*-means algorithm.

The results in Table III show that the variation of sensor to the quality of the classification performance. Compared with the classification rate of four sensors, those of six sensors are little bit improved as we expected although the difference is a little. It can be noted that the classification rate of the smell that can be controlled the quality of smell like JOSS and MOS can be perfectly classified in all cases. In the opposite way, the smell that has some fluctuation in the quality of smell tend to have low classification when the information of the input data is decreased. Therefore, to achieve the high classification results, it is important to note the quality of the tested smell.

B. Result for SMM

It seems that the results will be improved if we use many measurement data as much as possible. But there will be some measurement noises in the data, which make the results worse compared with the case that typical measurement data were used. Therefore, we try to reduce the noisy effect in the measurement data by using the regression model.

First, we divide the measurement data into n intervals. The measurement data in each segment are replaced by the linear regression model. We take $n = 3$ and the number of the data in each division becomes 20. Then we get the following linear regression model.

$$y_i = \beta_0 + \beta_1 x_i, i = 1, 2, ..., 20 \tag{6}$$

where y_i is the estimated voltage, $\beta_0$is the intercept and β_1 is the slope of the regression line, and x_i is the time. Then the slopes become

$$\beta_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{7}$$

where $\bar{x}$ and $\bar{y}$ are mean values of x_i and y_i, respectively.

The slope maximum mean(SMM) is to adopt the features of the data as five dimension(three slopes in each segment plus the maximum value and the mean value among the total data)as shown in Fig. 10. We use eight sensors as shown in Table I and the total number of input becomes forty(5 data times 8 sensors). For the hidden layer, we have tried several values, and the number of nodes which gives good accuracy and reasonable training time for both data turned out to be forty. The output layer contains twelve nodes, each node representing one data source. The learning rate, the momentum rate, and the minimum mean square error (MSE) during the training period are set by trial and error method to 0.1, 0.001, and 0.0003, respectively.

As the comparison of the features by SMM, we adopt the average of the TSD in the steady state, which means the maximum values among the total data.

Based on the information during the investigation of the correlation among data sets, most data sources are highly correlated to their repetition data with high SI values. Therefore, only one data set which has the highest average SI value to other repetition data sets from each source of fire are used as the training data for the BP, and the rest of the data are used as the test data. We assume that a pattern is classified correctly if either the output is not less than 0.7 and the target is 1, or the output is not greater than 0.3 and the target is 0.

For the *k*-means algorithm, the training data of the BP method are used as the initial data, and then the data patterns are assigned to the nearest cluster center according to the Euclidean distance. After that, the new cluster center is recalculated. The process continues until the position of the cluster center is not changed. The final results of this experiment are shown in Table III.

The results using the TSD from both the BP method and the *k*-means algorithm are better than those of the SSD case. The data signals from the MOGS are affected by many factors, such as the sampling condition, the inconsistency in burning rate, the fluctuation from the standard air, and so on. Therefore, the saturation stages of the data vary by those factors. By including the signal before approaching the saturation stage, the accuracy of classifying all the smells is increased.

C. Discussion

Although the distribution of PCA shown in Fig.6 cannot clearly separate similar sources of smell such as the aroma oil and the aroma candle, the BP method and the *k*-means algorithm are capable of classifying them perfectly as shown in Table III. The results of TSD using the BP method have only two incorrectly classified data. These two data are not misclassified as the other smells. Only the output value of their paper node are not high enough to classify them as the paper. The output values of these two data on the paper node are only 0.4951 and 0.4799 respectively, and the output of the other output nodes are nearly zero. The results are much better than those in [1], where two kinds of MOGS are used for classifying several sources of fire into three fire condition levels of flaming, smoldering, and nuisance, and its accuracy is only 85%. The smoke density of the tested data are not high enough to trigger the alarm of the smoke detector. In case of unusual burning smells in the residences such as the wood burning, flaming from the LPG, or the leakage of LPG, it is necessary to have a proper device to detect these sources before becoming unable to extinguish the fire. We can conclude that the new EN system shown in this paper is

TABLE V

RESULTS OF EXPERIMENT 1 TO SHOW THE EFFECT OF NUMBER OF SENSORS

Sources	**Four sensors data**				**Six sensor data**			
of	**BPNN**		***k*-means**		**BPNN**		***k*-means**	
fire	**Correct**	**%**	**Correct**	**%**	**Correct**	**%**	**Correct**	**%**
Steam	37/39	94.9	40/40	100	38/39	97.4	4040	100
Joss	39/39	100	40/40	100	39/39	100	40/40	100
Mos	39/39	100	40/40	100	39/39	100	40/40	100
Aroma	39/39	100	40/40	100	39/39	100	40/40	100
Candle	39/39	100	40/40	100	39/39	100	40/40	100
Flame	38/39	97.4	40/40	100	39/39	100	40/40	100
LPG	39/39	100	40/40	100	39/39	100	40/40	100
Oden	39/39	100	40/40	100	39/39	100	40/40	100
Oil	34/39	87.2	37/40	92.5	39/39	100	37/39	92.5
Toast	39/39	100	40/40	100	39/39	100	40/40	100
Paper	34/39	87.2	35/40	87.5	31/39	94.9	35/40	87.5
Wood	39/39	100	39/40	97.5	39/39	100	38/40	95
Average		97.2		98.1		99.4		98.5

TABLE VI

RESULTS OF EXPERIMENT II

Sources	**Full Time Series Data (TSD)**				**Saturation Stage Data(SSD)**			
of	**BP**		***k*-means**		**BP**		***k*-means**	
fire	**Correct**	**%**	**Correct**	**%**	**Correct**	**%**	**Correct**	**%**
Steam	39/39	100	40/40	100	38/39	97.4	39/40	97.5
Joss	39/39	100	40/40	100	39/39	100	40/40	100
Mos	39/39	100	40/40	100	39/39	100	40/40	100
Aroma	39/39	100	40/40	100	39/39	100	40/40	100
Candle	39/39	100	40/40	100	39/39	100	40/40	100
Flame	39/39	100	40/40	100	39/39	100	40/40	100
LPG	39/39	100	40/40	100	39/39	100	40/40	100
Oden	39/39	100	40/40	100	39/39	100	40/40	100
Oil	39/39	100	40/40	100	38/39	97.4	37/39	92.5
Toast	39/39	100	40/40	100	38/39	97.4	40/40	100
Paper	37/39	94.9	35/40	87.5	31/39	79.5	28/40	70
Wood	39/39	100	37/40	92.5	32/39	82.1	28/40	70
Average		99.6		98.3		96.2		94.2

a proper device for this application.

IX. CONCLUSION

We have presented the reliability of a new EN system designed from various kinds of MOGS. The EN has the ability to identify various sources of fire in the early stage with more than 99% of accuracy by using only a single training data set in the BP case. The results from the *k*-means algorithm also shows the ability to predict the sources of fire with more than 98% of accuracy. It can be concluded that the EN is suitable for detecting the early stage of fire.

ACKNOWLEDGMENT

The authors would like to thank Dr. Bancha Chrarumporn at Rajamangala University of Technology Lanna, Thailand and Mr. Nobuaki Murakami of FIS Inc. for the technical and experiment support to achieve the present paper.

REFERENCES

[1] N. Baric, M. Bucking, and M. Rapp, " A Novel Electronic Nose based on Minimized SAW sensor arrays coupled with SPME Enchanced Headspace-Analysis and its Use for Rapid Determination of volatile Organic Compounds in Food Quality Monitoring", *Sensors and Actuator B*, Vol. 114, pp. 482–488, 2006.

[2] M. Kusuke, A.C. Romain, and J. Nicolas, "Microbial Volatile Organic Compounds as Indicators of Fungi. Can an Electronic Nose Defect Fungi in Odoor Environments? ", *International Journal of Building Science and its Applications*, Vol. 40, pp. 213–222, 1995.

[3] A. Norman, F. Stam, A. Morrissey, M. Hirschfelder, and D. Enderlein, "Packaging Effects of a Novel Explosion-Proof Gas Sensor", *Sensors and Actuator B*, Vol. 95, pp. 287–290, 2003.

[4] R. C. Young, W. J. Buttner, B. R. Linnel, R. Ramesham, "Electronic Nose for Space Program Applications", *Sensors and Actuator B*, Vol. 93, pp. 7–16, 2003.

[5] J. A. Milke, "Application of Neural Networks for discriminating Fire Detectors", *International Conference on Automatic Fire Detection, AUBE'95, 10th, Duisburg, Germany*, pp. 213–222, 1995.

[6] B. Charumporn, M. Yoshioka, T. Fujinaka, and S. Omatu, "An Electronic Nose System Using Back Propagation Neural Networks with a Centroid Training Data Set", *Proc. Eighth International Symposium on Artificial Life and Robotics, Japan*, pp. 605–608, 2003.

[7] General Information for TGS sensors, *Figaro Engineering*, available at www.figarosensor.com/products/general.pdf.

[8] W. L. Carlson and B. Thorne, *Applied Statistical Methods*, Prentice Hall International, 1997.

[9] T. Fujinaka, M. Yoshioka, S. Omatu, and T. Kosaka, "Intelligent Electronic Nose Systems for Fire Detection Systems Based on Neural Networks", *The second International Conference on Advanced Engineering Computing and Applications in Sciences, Valencia, Spain*, pp.73–76, 2008.

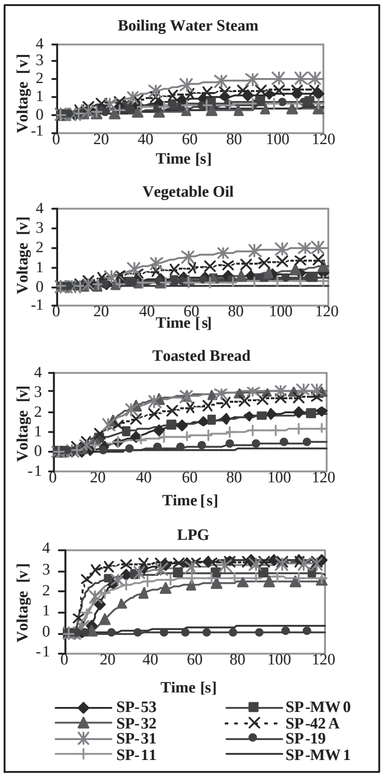
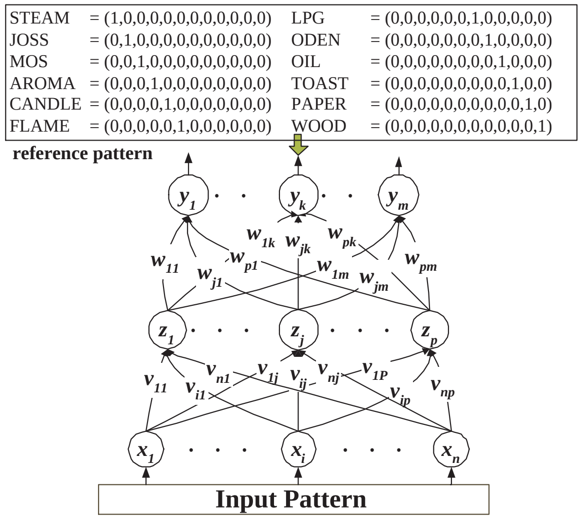


Fig. 7. Time series data from some sources of fire in the experiment



STEAM	= (1,0,0,0,0,0,0,0,0,0,0,0,0,0)	LPG	= (0,0,0,0,0,0,1,0,0,0,0,0,0)
JOSS	= (0,1,0,0,0,0,0,0,0,0,0,0,0,0)	ODEN	= (0,0,0,0,0,0,0,1,0,0,0,0,0)
MOS	= (0,0,1,0,0,0,0,0,0,0,0,0,0,0)	OIL	= (0,0,0,0,0,0,0,0,0,1,0,0,0)
AROMA	= (0,0,0,1,0,0,0,0,0,0,0,0,0,0)	TOAST	= (0,0,0,0,0,0,0,0,0,0,1,0,0)
CANDLE	= (0,0,0,0,1,0,0,0,0,0,0,0,0)	PAPER	= (0,0,0,0,0,0,0,0,0,0,0,1,0)
FLAME	= (0,0,0,0,0,1,0,0,0,0,0,0,0)	WOOD	= (0,0,0,0,0,0,0,0,0,0,0,0,1)

Fig. 9. Neural network for classification

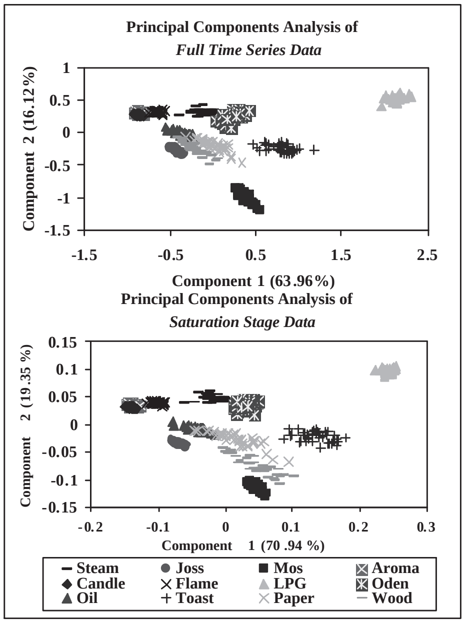


Fig. 8. Two main components of the experimental data using the PCA

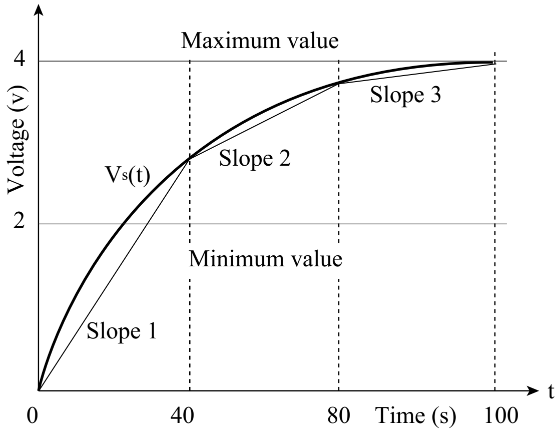


Fig. 10. Neural network for classification

iaria journals

www.iariajournals.org

International Journal On Advances in Intelligent Systems

- ICAS, ACHI, ICCGI, UBICOMM, ADVCOMP, CENTRIC, GEOProcessing, SEMAPRO, BIOSYSCOM, BIOINFO, BIOTECHNO, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS
- issn: 1942-2679

International Journal On Advances in Internet Technology

- ICDS, ICIW, CTRQ, UBICOMM, ICSNC, AFIN, INTERNET, AP2PS, EMERGING
- issn: 1942-2652

International Journal On Advances in Life Sciences

- eTELEMED, eKNOW, eL&mL, BIODIV, BIOENVIRONMENT, BIOGREEN, BIOSYSCOM, BIOINFO, BIOTECHNO
- issn: 1942-2660

International Journal On Advances in Networks and Services

- ICN, ICNS, ICIW, ICWMC, SENSORCOMM, MESH, CENTRIC, MMEDIA, SERVICE COMPUTATION
- issn: 1942-2644

International Journal On Advances in Security

- ICQNM, SECURWARE, MESH, DEPEND, INTERNET, CYBERLAWS
- issn: 1942-2636

International Journal On Advances in Software

- ICSEA, ICCGI, ADVCOMP, GEOProcessing, DBKDA, INTENSIVE, VALID, SIMUL, FUTURE COMPUTING, SERVICE COMPUTATION, COGNITIVE, ADAPTIVE, CONTENT, PATTERNS
- issn: 1942-2628

International Journal On Advances in Systems and Measurements

- ICQNM, ICONS, ICIMP, SENSORCOMM, CENICS, VALID, SIMUL
- issn: 1942-261x

International Journal On Advances in Telecommunications

- AICT, ICDT, ICWMC, ICSNC, CTRQ, SPACOMM, MMEDIA
- issn: 1942-2601