Kurt Bauknecht
Birgit Pröll
Hannes Werthner (Eds.)

LNCS 4082

E-Commerce and Web Technologies

7th International Conference, EC-Web 2006
Krakow, Poland, September 2006
Proceedings

Springer

Lecture Notes in Computer Science 4082

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Kurt Bauknecht Birgit Pröll
Hannes Werthner (Eds.)

E-Commerce and Web Technologies

7th International Conference, EC-Web 2006
Krakow, Poland, September 5-7, 2006
Proceedings

Springer

Volume Editors

Kurt Bauknecht
University of Zurich, Department of Informatics (IFI)
Winterthurer Strasse 190, 8057 Zurich, Switzerland
E-mail: baukn@ifi.unizh.ch

Birgit Pröll
Johannes Kepler University Linz, Institute for Applied Knowledge Processing (FAW)
Softwarepark Hagenberg, 4232 Hagenberg, Austria
E-mail: bproell@faw.uni-linz.ac.at

Hannes Werthner
Vienna University of Technology
Institute of Software Technology and Interactive Systems, EC-Group
Favoritenstr. 9-11, 1040 Vienna, Austria
E-mail: hannes.werthner@ec.tuwien.ac.at

Library of Congress Control Number: 2006931012

CR Subject Classification (1998): H.4, K.4.4, J.1, K.5, H.3, H.2, K.6.5

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-540-37743-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-37743-6 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11823865 06/3142 5 4 3 2 1 0

Preface

We welcome you to the 7th International Conference on E-commerce and Web Technologies (EC-Web 2006) held in Krakow, Poland, in conjunction with DEXA 2006. This conference was organized for the first time in Greenwich, UK, in 2000, and it has been able to attract an increasing number of participants and interest, reflecting the progress made in the field. As in the previous years, EC-Web 2006 served as a forum bringing together researchers from academia and practitioners from industry to discuss the current state of the art in e-commerce and Web technologies. Inspiration and new ideas emerged from intensive discussions that took place during the keynote address, the formal sessions and the social events.

The conference attracted 88 paper submissions and each paper was reviewed by three Program Committee members. The Program Committee selected 24 papers for presentation and publication (an acceptance and publication rate of 27%). We have to confess that this task was not that easy due to the high quality of the submitted papers.

We would like to express our thanks to our colleagues who helped put together the technical program: the Program Committee members and external reviewers for their timely and rigorous reviews of the papers, and the Organizing Committee for their help in the administrative work and support. We owe special thanks to Gabriela Wagner for her helping hand concerning the administrative and organizational tasks of this conference.

Finally, we would like to thank all the authors who have submitted papers, authors who presented papers, and the participants who made this conference an intellectually stimulating event.

We hope that all attendees enjoyed the hospitality of Krakow and the conference.

August 2006

Birgit Pröll (Johannes Kepler University of Linz, Austria)
Hannes Werthner (Vienna University of Technology, Austria)
Program Chairs EC-Web 2006

Organization

Program Committee

Marco Aiello, University of Trento, Italy
Sourav S. Bhowmick, Nanyang Technological University, Singapore
Martin Bichler, Technical University Munich, Germany
Susanne Boll, University of Oldenburg, Germany
Stephane Bressan, National University of Singapore, Singapore
Tanya Castleman, Deakin University, Australia
Wojciech Cellary, The Poznan University of Economics, Poland
Jen-Yao Chung, IBM T.J. Watson Research Center, USA
Alfredo Cuzzocrea, University of Calabria, Italy
Eduardo Fernandez, Florida Atlantic University, USA
Elena Ferrari, University of Insubria at Como, Italy
Farshad Fotouhi, Wayne State University, USA
Karl A. Fröschl, Electronic Commerce Competence Center, Austria
Yongjian Fu, Cleveland State University, USA
Stephane Gagnon, New Jersey Institute of Technology, USA
Fausto Giunchiglia, University of Trento, Italy
Chanan Glezer, Ben Gurion University, Israel
Thomas Hess, LMU Munich, Germany
Yigal Hoffner, Switzerland
Christian Huemer, University of Vienna, Austria
Gregory E. Kersten, Concordia University Montreal, Canada
Hiroyuki Kitagawa, University of Tsukuba, Japan
Gabriele Kotsis, Johannes Kepler University Linz, Austria
Alberto Laender, Federal University of Minas Gerais, Brazil
Juhnyoung Lee, IBM T. J. Watson Research Center, USA
Leszek Lilien, Western Michigan University, USA
Ee-Peng Lim, Nanyang Technological University, Singapore
Huan Liu, Arizona State University, USA
Heiko Ludwig, IBM T. J. Watson Research Center, USA
Sanjay Kumar Madria, University of Missouri-Rolla, USA
Bamshad Mobasher, DePaul University, USA
Natwar Modani, IBM India Research Lab, India
Mukesh Mohania, IBM India Research Lab, India
Guenter Mueller, University of Freiburg, Germany
Dirk Neumann, University of Karlsruhe, Germany
Gustaf Neumann, Vienna University of Economics and BA, Austria
Wee Keong Ng, Nanyang Technological University, Singapore
Rolf Oppliger, eSECURITY Technologies, Switzerland
Oscar Pastor, Valencia University of Technology, Spain
Guenther Pernul, University of Regensburg, Germany

Evangelia Pitoura, University of Ioannina, Greece
Ivana Podnar, EPFL, Switzerland
Giuseppe Psaila, University of Bergamo, Italy
Gerald Quirchmayr, University of Vienna, Austria
Indrakshi Ray, Colorado State University, USA
Werner Retschitzegger, Johannes Kepler University Linz, Austria
Tomas Sabol, Technical University of Kosice, Slovakia
Nandlal L. Sarda, Indian Institute of Technology Bombay, India
Steffen Staab, University of Koblenz, Germany
Michael Stroebel, BMW Group, Germany
Roger M. Tagg, University of South Australia, Australia
Kian-Lee Tan, National University of Singapore, Singapore
Stephanie Teufel, University of Fribourg, Switzerland
Bruce H. Thomas, University of South Australia, Australia
A Min Tjoa, Technical University of Vienna, Austria
Aphrodite Tsalgatidou, University of Athens, Greece
Krishnamurthy Vidyasankar, Memorial University of Newfoundland, Canada
Hans Weigand, Tilburg University, The Netherlands
Christof Weinhardt, University of Karlsruhe, Germany
Janusz Wielki, Technical University of Opole, Poland

External Reviewers

Nitin Agarwal, Arizona State University, USA
George Athanasopoulos, University of Athens, Greece
Michael Borovicka, University of Innsbruck, Austria
Peter Butka, Technical University of Kosice, Slovakia
Karol Furdik, Intersoft, a.s., Slovakia
Wojciech Galuba, EPFL, Switzerland
Jörg Gilberg, University of Regensburg, Germany
Christoph Grün, Vienna University of Technology, Austria
Fabius Klemm, EPFL, Switzerland
Jan Kolter, University of Regensburg, Germany
Marian Mach, Technical University of Kosice, Slovakia
Patrick Sinclair Merten, University of Fribourg, Switzerland
Sai Moturu, Arizona State University, USA
Björn Muschall, University of Regensburg, Germany
Michael Pantazoglou, University of Athens, Greece
Marek Paralic, Technical University of Kosice, Slovakia
Lance Parsons, Arizona State University, USA
Vicente Pelechano, Valencia University of Technology, Spain
Gonzalo Rojas, Valencia University of Technology, Spain
Marta Ruiz, Valencia University of Technology, Spain
Jarogniew Rykowski, Poznan University of Economics, Poland
Ali Salehi, EPFL, Switzerland
Martin Steinert, University of Fribourg, Switzerland

Sergiusz Strykowski, Poznan University of Economics, Poland
Lei Tang, Arizona State University, USA
Victoria Torres, Valencia University of Technology, Spain
Pedro Valderas, Valencia University of Technology, Spain
Le-Hung Vu, EPFL, Switzerland
Daniela Wanner, University of Fribourg, Switzerland
Marco Zapletal, Vienna University of Technology, Austria
Zheng Zhao, Arizona State University, USA

Table of Contents

Recommender System 1

Recommender Systems 2

Business Process / Design Aspects

Mobile Commerce

Security and E-Payment

Web Services Computing / Semantic Web

E-Negotiation and Agent Mediated Systems

Issues in Web Advertising

Map-Based Recommendation of Hyperlinked Document Collections

Mieczysław A. Kłopotek, Sławomir T. Wierzchoń, Krzysztof Ciesielski,
Michał Dramiński, and Dariusz Czerski

Institute of Computer Science, Polish Academy of Sciences,
ul. Ordona 21, 01-237 Warszawa, Poland
{kciesiel, klopotek, mdramins, stw, dcz}@ipipan.waw.pl

Abstract. The increasing number of documents returned by search engines for typical requests makes it necessary to look for new methods of representation of the search results.

In this paper, we discuss the possibility to exploit incremental, navigational maps based both on page content, hyperlinks connecting similar pages and ranking algorithms (such as HITS, SALSA, PHITS and PageRank) in order to build visual recommender system. Such system would have an immediate impact on business information management (e.g. CRM and marketing, consulting, education and training) and is a major step on the way to information personalization.

1 Introduction

Recommender systems became indispensable part of modern e-business, especially using the electronic medium of the Internet. It is claimed that even 20% of clients may be encouraged to purchase a good by the most successful recommender systems. Recommenders are applied in advertisement of books (Amazon), CD's (MediaUnbound, MoodLogic, CDNow, SongExplorer), films (Reel.com Movie Matcher, MovieFinder.com Mach Maker), cosmetics (Drugstore.com) and other. But also recommender systems are applied when advice is sought by firms when selecting training courses for employees, job search for unemployed etc.

Recommender systems are not only applied to increase sales, but also to expand cross-selling, attraction of new clients, extending of trust of existent clients, to overcome the barriers of mass advertisement through high personalization of offers. The intense research did not, however, overcome many important limitations, like sparseness of data [21], scalability and real time action, detail level of available data, feedback to the sales firm [22], protection against preference manipulation, visualization of recommendation acceptability reasons [15], client modeling, system evaluation, creation of distributed systems [3], etc.

In our on-going research effort, we target at recommender systems capable of overcoming the limitations of present-day systems with respect to problems of rare data in recommendation, scalability and visualization of recommendation

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 1–10, 2006.

for the purpose of proper explanation and justification of recommendation. Results of this research will surely lead to practical guidelines for construction of commercial recommender systems, where above mentioned problems are crucial.

We have created a full-fledged search engine BEATCA for small collections of documents (up to several millions) capable of representing on-line replies to queries in graphical form on a document map. We extended WebSOM's goals by a multilingual approach, new forms of geometrical representation and we experimented also with various modifications to the clustering process itself [17,18] . The crucial issue for understanding the 2D map by the user is the clustering of its contents and appropriate labeling of the clustered map areas.

Several important issues need to be resolved in order to present the user with an understandable map of documents. The first issue is the way of document clustering. In the domains, like e.g. legal documents, where the concepts are not sharply separated, a fuzzy-set theoretic approach to clustering appears to be a promising one. The other one is the issue of initialization of topical maps. Our experiments showed that the random initialization performed in the original WebSOM may not lead to appearance of meaningful structure of the map. Therefore, we proposed several methods for topical map initialization, based on SVD, PHITS, PLSA and Bayesian network techniques.

In this paper, before we report on our current state of research effort starting with Sect. 4, we briefly present an overview of recommender system concepts (Sect. 2) and our concept of an integrated recommender system (Sect. 3).

2 Recommender Systems Overview

Intelligent agent (IA) is a user's assistant or recommender system based on machine learning and data mining techniques. Construction of IA uses the following paradigm taken from the ordinary life: "people uses helpful information without any fixed plans". People do not need and cannot describe their own work in terms of coefficients and classification. People just operate and know what they are interesting in, or what they want when they see it.

Recommender system simulates some social behavior. No one has unlimited knowledge and such knowledge is not necessary on daily basis. However, in some decision problems we have to go into details of specific, narrow knowledge. Sometimes there is a possibility to use advice from experienced person (expert) in a given area. Recommender systems try to help user in such situation by using knowledge collected in specified discipline and watching decisions made by other users decisions in the similar case. These systems have been built as a help in decision process for people, but also for multi-agent systems and generally speaking for systems consisting of objects that have limited knowledge about environment. Recommender system uses knowledge about passive objects to recommend next (somehow similar) item to active objects. For example, recommender system can recommend next web page or article in Internet shop (passive object) that user (active object) is probably looking for.

Recommender systems may be classified along the following criteria: amount and type of data that come from active object, amount and type of required data about community of active objects, method of recommendation, result type of recommendation, way of delivering recommendation to the active object and the degree of personalization (adaptivity to active object characteristic). More detailed classification and examples of commercial recommender systems can be found in [22].

Methods of recommendation in early systems were based mostly on the following approaches: recommendations based on searching, categories, clustering, association rules or classifiers. Finally, evolution of recommender systems has led to two major approaches in construction of IA:

1. Content-based approach. System creates users profiles by analyzing their operations and recommends documents that are compatible with these profiles.
2. Collaborative approach - collaborative or social filtering [12]. System focuses on a group of users.

The first approach, rooted in the tradition of information processing, is applicable if the system deals with text only. The system seeks information similar to that preferred by the user. If a user is interested in some knowledge areas (represented by documents described by some keywords or phrases) then the recommender looks for documents with similar content to already articulated. The basic problem here is to capture all specific aspects of a document content (e.g. in disciplines such as music, film, computer-related issues etc.). Even restricting recommendations to text documents only, most representations are able to cover only some aspects of document content, which results in weak quality of presented recommendations.

The second approach, called also social learning, relies on exploiting reactions of other users to the same object (e.g. a course, educational path, a film, etc.). The system looks for users with similar interests, capabilities etc. and recommends them information or items they are searching for.

This approach allows for posing questions like "show me information I have never seen but it turned interesting to people like me". Personalized information is provided in an iterative process where information is presented and user is asked to rank it, what allows to determine his/her profile. This profile is next used to locate other users with similar interests, in order to identify groups with similar interests.

Instead of calculating similarity between documents, this method determines degree of membership to a group (for example based on surveys). In contrast to first approach, it does not require analysis of document content, what means that document with arbitrary content could be presented to the user, with the same probability. Each document is assigned with identifier and a degree of membership to a group.

This approach is characterized by two features: first of all the document relevance is determined in the context of the group and not of a single user. Second, evaluation of the document is subjective. Hence one can handle complex and heterogeneous evaluation schemas.

3 Integrated Recommendations

Existing recommender systems, based on a paradigm of content-based filtering as well as those based on collective filtering principle, do not take into consideration possible synergic effects. Such effects emerge when:

- both methodologies are merged,
- system is able to model joint, integrated recommendation of passive and active objects (i.e. clients and products), and not only passive objects pointed by active ones,
- recommendations are based on visual system, which helps to explain and justify a recommendation.

Application of joint methodology is possible if available data contain information on recommended objects as well as relations between recommended and recommending objects. Such information is present, e.g. in WWW documents, where individual html pages have not only textual context, but also hyperlinks between them. From logs saved on a particular host one can obtain so-called click-stream of users surfing from one page to another, and some additional data such as voluntarily filled-in questionnaires. Among other examples are libraries, book stores, or any shop (including e-shops), where products can be described by a set of attributes (e.g. advertisement leaflet) and users can be identified by some ID cards (e.g. loyalty program participation cards). Similarly, for some services (e.g. concerning education or health), both pointed(passive) and pointing(active) objects are described by attributes.

By an *integrated recommendation* we mean recommendation such as "People interested in <characteristics of people> are buying also book <title>" (instead of typical recommendation in form: "People interested in <title> are buying also book <title>"). Thus, integrated recommendation requires that system has an ability to generalize features describing characteristics of active objects (i.e. users or clients).

Recommendation with a visual explanation and justification is a completely new approach, based on creation of two-dimensional, navigational map of objects. Such a map yields a possibility to present an identified area of user's interests together with surrounding context, i.e. main directions of his/her future activities.

4 BEATCA Search Engine

Our first step towards a new model of recommendation system was to create a new-type search engine, based on a document map interface. Our map-based approach to search engine interfacing comprises two important features from the point of view of the target recommendation system: providing an overview over the whole collection of objects, and a very detailed clustering into groups of objects and their immediate (local) contexts.

With a strongly parameterized map creation process, the user of BEATCA can accommodate map generation to his particular needs, or even generate multiple maps covering different aspects of document collection. The overall complexity of the map creation process, resulting in long run times, as well as the need to avoid "revolutionary" changes of the image of the whole document collection, require an incremental process of accommodation of new incoming documents into the collection.

Within the BEATCA project we have devoted much effort to enable such a gradual growth. In this study, we investigated vertical (new topics) and horizontal (new documents on current topics) growth of document collection and its effects on the map formation capability of the system. To ensure intrinsic incremental formation of the map, all the computation-intense stages involved in the process of map formation (crawling, indexing, GNG clustering, SOM clustering) need to be reformulated in terms of incremental growth.

In particular, Bayesian Network driven crawler is capable of collecting documents around an increasing number of distinct topics. The crawler learning process runs in a kind of horizontal growth loop while it keeps its performance with increasing number of documents collected. It may also grow vertically, as the user can add new topics for searching.

In the next section we briefly mention our efforts to create a crawler, that can collect documents from the internet devoted to a selected set of topics. The crawler learning process runs in a kind of horizontal growth loop while it improves its performance with increase of the amount of documents collected. It may also grow vertically, as the user can add new topics of for search during its run time.

4.1 Intelligent Topic-Sensitive Crawling

The aim of intelligent crawling [1] is to crawl efficiently documents belonging to certain topics. Often it is particularly useful not to download each possible document, but only that which concerns a certain subject. In our approach we use Bayesian nets (BN) and HAL algorithm to predict relevance of documents to be downloaded.

Topic-sensitive crawler begins processing from several initial links, specified by the user. To describe a topic of our interest, we use query document. This special pseudo document contains descriptive terms with a priori given weights, which are later used to calculate priorities for crawled documents. During crawling first few hundred documents, crawler behavior depends only on initial query. Later the query is expanded by BN or HAL methods described below.

4.1.1 Bayesian Net Document Query Expansion

At increasing time intervals, we build Bayesian Net by using ETC learning algorithm [16] to approximate term co-occurence in topical areas. We use them to expand query and to calculate priorities for further documents links. We expand query by adding parent and children nodes of BN terms, which are already present in query document. New terms get weights proportional to the product of the likelihood of their co-occurrence and the weight of the original term.

4.1.2 HAL Document Query Expansion

To expand query document we also use HAL (*Hyperspace Analogue To Language*, [20]) model. It is based on psychological theory claiming that meaning of a word is a function of contexts in which it appears; and the words sharing contexts have similar meanings. From computational perspective, HAL model can be represented as a matrix H in which cell h_{ij} corresponds to similarity measure of terms i and j.

Like in the BN algorithm, final document links priorities are calculated by modified cosine measure between new expanded query document and document containing those links.

4.1.3 Evaluation

To see, how effective the proposed topic sensitive crawkling is, We run two experiments, one for BN algorithm, the other for HAL algorithm [5]. In both cases, three seed links [`http://java.sun.com/j2ee/index.jsp`, `http://java.sun.com/products/ejb/`, `http://www.javaskyline.com/learning.html`] served as starting points. We used a query consisting of six weighed descriptive terms, [java(with weight of 20) documentation(30) ejb(100) application(50) server(50) J2EE(30)]. Figure 1(a) depicts results for crawler based on BN algorithm and figure 1(b) presents results for crawler based on HAL algorithm.

Quality measure is the average relevance measure, computed after every 500 new document downloads. Relevance is equal to modified cosine measure, but only for terms which are present in the initial user query ($Q = Q_0$), i.e. $relevance = cos(q_0, d)$.

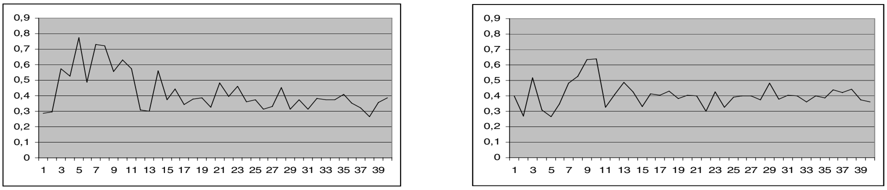


Fig. 1. Crawler evaluation (20000 documents downloaded): (a) Bayesian Net algorithm (b) HAL algorithm

Both methods appear to be satisfactory: average cosine measure amounts 0.4. The crawler does not lose a priori defined topic during the crawl. BN proved to be faster of the two methods, but it requires to stop whole process in order to rebuild BN model. HAL table can be built during the crawl, but it requires more computations.

4.2 Map Creation Process Outline

4.2.1 WebSOM Approach

One of main goals of the project is to create 2D document map in which geometrical vicinity would reflect conceptual closeness of documents in a given document set. Additional navigational information (based on hyperlinks between documents) is introduced to visualize directions and strength of between-group topical connections. Our starting point was widely-known Kohonen's Self-Organizing Map principle [19], which is an unsupervised learning neural network model, consisted of regular, 2D grid of neurons.

4.2.2 Growing Neural Gas Approach

Similarly to WebSOM, growing neural gas (GNG) can be viewed as topology learning algorithm, i.e. its aim is to find a topological structure which closely reflects the topology of a given collection of high-dimensional data. In typical SOM the number of units and topology of the map is predefined. As observed in [10], the choice of SOM structure is difficult, and the need to define a decay schedule for various features is problematic.

GNG starts with very few units and new units are inserted successively every each few iterations. To determine where to insert new units, local error measures are gathered during the adaptation process; new unit is inserted near the unit, which has accumulated maximal error. Interestingly, GNG cells of the GNG network are joined automatically by links, hence as a result a possibly disconnected graph is obtained, and its connected components can be treated as different data clusters. The complete GNG algorithm details and its comparison to numerous other soft competitive methods can be found in [11].

4.2.3 GNG with Utility Factor

Typical problem in web mining applications is that processed data is constantly changing - some documents disappear or become obsolete, while other enter analysis. All this requires models which are able to adapt its structure quickly in response to non-stationary distribution changes. Thus, we decided to implement and use GNG with utility factor model, presented by Fritzke in [11].

A crucial concept here is to identify the least useful nodes and remove them from GNG network, enabling further node insertions in regions where they would be more necessary. The utility factor of each node reflects its contribution to the total classification error reduction. In other words, node utility is proportional to expected error growth if the particular node would have been removed. There are many possible choices for the utility factor. In our implementation, utility update rule of a winning node has been simply defined as $U_s = U_s + error_t - error_s$, where s is the index of the winning node, and t is the index of the second-best node (the one which would become the winner if the actual winning node would be non-existent). Newly inserted node utility is arbitrarily initialized to the mean of two nodes which have accumulated most of the error: $U_r = \frac{U_u + U_v}{2}$.

After utility update phase, a node k with the smallest utility is removed if the fraction $\frac{error_j}{U_k}$ is greater then some predefined threshold; where j is the node with the greatest accumulated error. Detailed description of the GNG-U algorithm can be found in [11].

4.2.4 GNG Network Visualization

Despite many advantages over SOM approach, GNG has one serious drawback: high-dimensional networks cannot be easily visualized. Nevertheless, instead of single documents, we can build Kohonen map on GNG nodes reference vectors, treating each vector as a centroid representing a cluster of documents. Such a map is initialized in the same way as underlying GNG network (i.e. with the same broad topics) and next is learned in the usual manner. The resulting map is a visualization of GNG network with the detail level depending on the SOM size (since a single SOM cell can gather more than one GNG node). User can access document content via corresponding GNG node, which in turn can be accessed via SOM node - interface here is similar to the hierarchical SOM map case.

4.2.5 PHITS Technique

Alternatively to content-only based representation one can build a map which will visualize a model of linking patterns. In such model, document is represented as sparse vector, whose i-th component equals to path length *from* [via outcoming links] or *to* [via incoming links] to i-th document in a given collection. It is usually assumed, that these computations can be restricted to the paths of maximum length **5** and above this value document similarities are insignificant. It is also possible to estimate a joint term-citation model.

PHITS algorithm [6] does the same with link information as PLSA algorithm [13] with terms contained in a document. From mathematical point of view, PHITS is identical to PLSA, with one distinction: instead of modeling the citations contained within a document (corresponding to PLSA modeling of terms in a document), PHITS models ”in-links,” the citations to a document. It substitutes a citation-source probability estimate for PLSA term probability estimate. On the Web and in other document collections, usually both links and terms could or should be used for document clustering. The mathematical similarity of PLSA and PHITS enables to create a joint clustering algorithm [6], taking into consideration both similarity based on document content and citation patterns.

5 Final Remarks

Modern man faces a rapid growth in the amount of written information. Therefore he needs a means of reducing the flow of information by concentrating on major topics in the document flow. In order to achieve this, he needs a suitable recommendation system.

Grouping documents based on similar contents may be helpful in this context as it provides the user with meaningful classes or clusters. Document clustering and classification techniques help significantly in organizing documents in this way. A prominent position among these techniques is taken by the WebSOM of Kohonen and co-workers [19]. However, the overwhelming majority of the existing document clustering and classification approaches rely on the assumption that the particular structure of the currently available static document collection will not change in the future. This seems to be highly unrealistic, because both the interests of the information consumer and of the information producers change over time.

A recent study described in [14] demonstrated deficiencies of various approaches to document organization under non-stationary environment conditions of growing document quantity. The mentioned paper pointed to weaknesses among others of the original SOM approach (which itself is adaptive to some extent) and proposed a novel dynamic self-organizing neural model, so-called Dynamic Adaptive Self-Organising Hybrid (DASH) model. Other strategies like that of [9], attempt to capture the move of topics, enlarge dynamically the document map (by adding new cells, not necessarily on a rectangle map).

We take a different perspective in this paper claiming that the adaptive and incremental nature of a document-map-based search engine cannot be confined to the map creation stage alone and in fact engages all the preceding stages of the whole document analysis process.

Though one could imagine that such an accommodation could be achieved by "brute force" (learning from scratch whenever new documents arrive), there exists a fundamental technical obstacle for such a procedure: the processing time. The problem is even deeper and has a "second bottom": the clustering methods like those of SOM contain elements of randomness so that even re-clustering of the same document collection may lead to changes in resulting map.

The important contribution of our research effort so far is to demonstrate, that the whole incremental machinery not only works, but it works efficiently, both in terms of computation time, model quality and usability. . At the same time, it comes close to the speed of local search and is not directly dependent on the size of the model. This means that it is possible to target at large scale recommendation systems with a visual map-based interface.

Our investigation into influence of crawling via an intelligent crawling agent on the quality of the created document maps indicates a positive impact of this type of crawler on the overall map quality. This, together with known investigations of combined PLSA/PHITS model, seems to be an encouraging confirmation of our assumption that combining content-based and collaborative filtering may provide foundations for a more reliable recommendation.

Also apparently the new methods for creation of stable maps, that we propose, are successful to the extent that we may be able to develop visual recommendation justification in which changes in visual patterns will be attributed to real changes of user preferences and not due to artifacts of map construction algorithms.

References

1. C.C. Aggarwal, F. Al-Garawi, P.S. Yu: Intelligent crawling on the World Wide Web with arbitrary predicates. In Proc. 10th Int. World Wide Web Conference, pp. 96–105, 2001.
2. J.S. Breese, D. Heckerman, and D.C. Kadie: Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, 1998, pp. 43-52.

3. J. Callan, et. al: Personalisation and recommender systems in digital libraries, Joint NSF-EU DELOS Working Group Report, May 2003 `http://www.dli2.nsf.gov/internationalprojects/working_group_reports/personalisation.html`
4. S. Cayzer, U. Aickelin: A Recommender System based on Idiotypic Artificial Immune Networks, J. of Mathematical Modelling and Algorithms, 4(2)2005, 181-198
5. K. Ciesielski et al.: Adaptive document maps. In: Proc. IIPWM'06, Springer.
6. D. Cohn, H. Chang: Learning to probabilistically identify authoritative documents, Proceedings of the 17th International Conference on Machine Learning, 2000
7. M.W. Berry: Large scale singular value decompositions, Int. Journal of Supercomputer Applications, 6(1), 1992, pp.13-49
8. R. Decker: Identifying patterns in buying behavior by means of growing neural gas network, Operations Research Conference, Heidelberg, 2003
9. M. Dittenbach, A. Rauber, D. Merkl: Discovering hierarchical structure in data using the growing hierarchical Self-Organizing Map, Neurocomputing, 48 (1-4)2002, pp. 199-216
10. B. Fritzke: A growing neural gas network learns topologies, in: G. Tesauro, D.S. Touretzky, and T.K. Leen (Eds.) Advances in Neural Information Processing Systems 7, MIT Press Cambridge, MA, 1995, pp. 625-632
11. B. Fritzke, A self-organizing network that can follow non-stationary distributions, in: Proc. of the Int. Conference on Artificial Neural Networks '97, 1997, 613-618
12. D. Goldberg, D. Nichols, B.M. Oki, D. Terry: Using collaborative filtering to weave an information tapestry, Communication of the ACM, 35:61-70, 1992.
13. T. Hoffmann: Probabilistic latent semantic analysis, in: Proceedings of the 15th Conference on Uncertainty in AI, 1999
14. C. Hung, S. Wermter: A constructive and hierarchical self-organising model in a non-stationary environment, Int. Joint Conference in Neural Networks, 2005
15. A. Jameson: More than the sum of its Mmmbers: Challenges for group recommender. Proc. of the Int. Working Conference on Advanced Visual Interfaces, Gallipoli, Italy, 2004 `http://dfki.de/~jameson/pdf/avi04.jameson-long.pdf`
16. M. Kłopotek: A new Bayesian tree learning method with reduced time and space complexity, Fundamenta Informaticae, 49(4) 2002, IOS Press, pp. 349-367
17. M. Kłopotek, M. Dramiński, K. Ciesielski, M. Kujawiak, S.T. Wierzchoń: Mining document maps, in Proceedings of Statistical Approaches to Web Mining Workshop (SAWM) at PKDD'04, M. Gori, M. Celi, M. Nanni eds., Pisa, 2004, pp.87-98
18. M. Kłopotek, S.T. Wierzchoń, K. Ciesielski, M. Dramiński, M. Kujawiak: Coexistence of fuzzy and crisp concepts in document maps, in: Proc. of the Int. Conference on Artificial Neural Networks (ICANN 2005), LNAI 3697, Springer-Verlag, 2005
19. T. Kohonen, Self-Organizing Maps, Springer Series in Information Sciences, vol. 30, Springer, 2001
20. B. C., K. Livesay, and K. Lund. Explorations in context space: Words, sentences, discourse. Discourse Processes, 25(2-3):211–257, 1998
21. Sarwar, G. Karypis, J. Konstan, J. Riedl: Item-based Collaborative Filtering Recommendation Algorithms, WWW10, May 1-5, 2001, Hong Kong
22. J. B. Schafer, J. Konstan, J. Riedl: Electronic Commerce Recommender Applications, Journal of Data Mining and Knowledge Discovery, 5(1-2): 115–152, 2001
23. U. Shardanand and P. Maes. Social information filtering: algorithms for automating "word of mouth". In ACM Conference Proceedings on Human Factors in Computing Systems, pages 210–217, Denver, CO, May 7-11 1995.

Web User Segmentation Based on a Mixture of Factor Analyzers

Yanzan Kevin Zhou[1] and Bamshad Mobasher[2]

[1] eBay Inc., San Jose, CA
yanzzhou@ebay.com
[2] DePaul University, Chicago, IL
mobasher@cs.depaul.edu

Abstract. This paper proposes an approach for Web user segmentation and online behavior analysis based on a mixture of factor analyzers (MFA). In our proposed framework, we model users' shared interests as a set of common latent factors extracted through factor analysis, and we discover user segments based on the posterior component distribution of a finite mixture model. This allows us to measure the relationships between users' unobserved conceptual interests and their observed navigational behavior in a principled probabilistic manner. Our experimental results show that the MFA-based approach results in finer-grained representation of user behavior and can successfully discover heterogeneous user segments and characterize these segments with respect to their common preferences.

1 Introduction

Web sites are increasingly becoming more complex, often involving a myriad of functions and tasks that can be performed online, or diverse content areas that span a variety of topics or subtopics. Therefore, increasingly sophisticated models are necessary to precisely capture Web user's interests and preferences. *Web usage mining* [1,10,6] plays a key role in Web user modeling. It is the most direct approach to studying Web users' online behavior since its primary data source is clickstream data, which is generated by Web users' interaction with a Web site and recorded in application or Web server log files. A variety of data mining and statistical techniques have been applied to discover useful patterns, such as Web page association rules [7,4] or user clusters [8].

In particular, user segmentation is a widely used approach for characterizing and understanding user behavior and interests in a Web site. Both distance-based clustering methods, such as k-means, and probabilistic density-based mixture models can be used for market segmentation purposes [11]. Clustering methods find groups of data objects based on their distances or distribution similarity. A disadvantage of distance-based methods is that no probabilistic inference can be made and, for high-dimensional data, the distance computation can be prone to noise and outliers. This can be partially remedied by model-based clustering,

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 11–20, 2006.

in which a mixture of probabilistic distributions are assumed to have generated the data, and the data objects that follow the same distribution can be regarded as a cluster. However, ordinary mixture models such as a Mixture of Gaussians (MoG) still suffer parameter over-fitting problems emanating from high-dimensional feature space. Furthermore, standard mixture models cannot discover the latent dimensional structure of the observed data which can "explain" the relationships among data objects.

Well-established latent variable models, such as Principal Component Analysis (PCA) and Factor Analysis (FA), are generally used for dimensionality reduction and discovery of latent structures in data. While the commonly used PCA method assumes zero-noise model, FA-based approaches distinguish between common variance and noise variance for each of the observed variables. Such noise discrimination effect is particularly important in the typically noisy Web navigation data [12].

In this paper we propose an approach for Web user segmentation and online behavior analysis based on a *Mixture of Factor Analyzers* (MFA). MFA is natural integration of finite mixture models and factor analysis, resulting in a statistical method which concurrently performs clustering and, within each cluster, local dimensionality reduction. This presents several benefits over approaches in which clustering and dimensionality reduction are performed separately. First, different features may be correlated within different clusters and thus the metric for dimensionality reduction may need to vary between different clusters. Conversely, the metric induced in dimensionality reduction may guide the process of cluster formation, i.e. different clusters may appear more separated depending on the local metric [3].

MFA has been shown effective in simultaneous case-space clustering and feature-space dimensionality reduction for the purpose of speech recognition and face detection [9,2]. However, to the best of our knowledge, MFA has not been used in modeling of Web user navigational patterns. Web usage data tends to be high dimensional, and Web users generally have different navigational behaviors based on their intended tasks or information needs, however, with common subpatterns, resulting in noisy patterns corresponding to multiple modalities. This, we believe, makes MFA particularly useful in Web user modeling: the mixture component variables model the global variation among individual Web users, and the latent dimensions in the factor analysis model allow for the conceptual representation of user's hidden interests without the typical noise. The discovered patterns are not only useful for online user behavior understanding, but also for other important e-commerce applications such as collaborative recommendation.

The paper is organized as follows. In Section 2 we discuss our MFA-based approach to model the multimodal Web navigation data and quantify the relationship between Web users' latent interests, segment memberships, and their observed behavior in user sessions. In Section 3 we introduce our approach for user segmentation based on posterior latent variable distribution in MFA. Section 4 presents our empirical results and verification based on experimental study of online user behavior on real world Web usage data.

2 Mixture of Factor Analyzers for Web Usage Data

We assume that appropriate preprocessing (such as data cleaning, sessionization, spider removal, etc.) has been performed on raw Web server logs [1], resulting in a set of p pages $\mathcal{D} = \{D_1, D_2, \cdots, D_p\}$ and a set of n user sessions $\mathcal{U} = \{U_1, U_2, \cdots, U_n\}$. Each user session can be represented as a p-dimensional vector $\mathbf{u} \in \mathbb{R}^p$ as $\mathbf{u} = [d_1, d_2, \cdots, d_p]^T$, representing user-page observations in that session, where the value of d_j is a weight associated with page D_j in the session (in our experiments, the weights are a function of the time spent on each page during the session).

In this section, we first present the basic elements of factor analysis for modeling Web navigational patterns, and then, we present our framework for extending the standard factor analysis model to a mixture model.

2.1 Using Factor Analysis to Model Web Usage Patterns

In standard maximum likelihood factor analysis (FA), an n-dimensional real-valued data vector $\mathbf{u}$ (in our case, a user session) is modeled using a k-dimensional vector of real-valued factors, $\mathbf{z}$, where k is generally much smaller than n. Since, in usage data these factors closely correspond to aggregate common interests of users, we call $\mathbf{z}$ a *preference vector*. Specifically, given $\mathbf{z} = [z_1, z_2, \cdots, z_k]^T \in \mathbb{R}^k$, we can view a user's access to a page D_i during a session as the sum of combined "influences" by the set of latent variables, each representing an abstract common preference. In other words, $d_i = l_{i1}z_1 + l_{i2}z_2 + ... + l_{ik}z_k + \epsilon_i$, where coefficient l_{ij} indicates how strongly the page D_i is related to user's preference Z_j, and ϵ_i represents independent random variance (noise) that is not accounted for by the k latent variables (which account for common variances). Since $k \ll p$, the original high dimensional data (at the page level) is mapped to a much lower-dimensional latent space with unwanted information modeled as random noise.

We can arrive at the density-based factor analysis model by defining proper probabilistic density functions (PDF) over the latent variables, and assuming a generative model for Web user session observations:

$$p(\mathbf{u}) = \int_{\mathbf{z}} p(\mathbf{u}|\mathbf{z})p(\mathbf{z})d\mathbf{z}, \quad \mathbf{z} \in \mathbb{R}^k. \tag{1}$$

We also obtain a linear Gaussian factor analysis model with the assumptions of a multivariate Gaussian prior over the latent variables $\mathbf{z}$, and an independent Gaussian noise model over ϵ, i.e.,

$$\mathbf{u} = \mathbf{L}\mathbf{z} + \epsilon; \quad \mathbf{z} \sim \mathcal{N}_k(0, \mathbf{I}), \quad \epsilon \sim \mathcal{N}_p(0, \mathbf{\Psi}), \tag{2}$$

where $\sim \mathcal{N}_k(0, \mathbf{I})$ denotes a k-variate joint Gaussian with zero mean and identity covariance matrix, and $\mathbf{\Psi}$ is the diagonal covariance matrix of random variances $diag(\sigma_i^2)$. Then, based on Equation 1 we can derive the unconditional PDF for the data, $p(\mathbf{u}) = \mathcal{N}_p(0, \mathbf{L}\mathbf{L}^T + \mathbf{\Psi})$.

2.2 MFA-Based Web User Modeling

Conceptually, We regard a user's online navigation behavior as the following generative process:

1. When a user u comes to a Web site, the user is assigned to a certain activity group g based on his current browsing history.
2. The expected activity of the user in group g is then determined by his k-dimensional preference vector.

Mathematically, a mixture model is a linear combination of m mixture component densities weighted by their prior probabilities (mixing proportion), i.e.,

$$p(\mathbf{u}) = \sum_{g=1}^{m} p(\mathbf{u}, g) = \sum_{g=1}^{m} p(\mathbf{u}|g)P(g) \tag{3}$$

where g is a discrete mixture variable taking value $g \in \{1, 2, \cdots, m\}$ and satisfying the condition $\sum_{g=1}^{m} P(g) = 1, P(g) \geq 0$.

If in the mixture model (Equation 3), each class-conditional density is a latent variable model (equation 1), then we obtain a *mixture of latent variable models.* In our case, we obtain a mixture of factor analyzers. The marginal observation distribution is obtained by integrating over both discrete mixture variable g and continuous latent variables $\mathbf{z}$

$$p(\mathbf{u}) = \sum_{g=1}^{m} \int_{\mathbf{z}} p(\mathbf{u}|\mathbf{z}, g)p(\mathbf{z})P(g)d\mathbf{z} \tag{4}$$

resulting in the following concrete unconditional PDF for MFA:

$$p(\mathbf{u}) = \sum_{g=1}^{m} \mathcal{N}_p(\mu_g, \mathbf{L}_g\mathbf{L}_g^T + \boldsymbol{\Psi})P(g) \tag{5}$$

where each mixture component has its own mean μ_g and loading matrix $\mathbf{L}_g$ together with its prior probability $P(g)$. This allows each factor analyzer to model the data covariance structure in a different part of the input space. The estimation of MFA parameters $\mu_g, \mathbf{L}_g, \boldsymbol{\Psi}_g, \{P(g)|g \in [1, m]\}$, is achieved by the EM algorithms, which is commonly used for latent variable models. Interested readers can refer to [3] and [5] for more details.

MFA is a nonlinear extension of linear Gaussian FA addressing both global data modal heterogeneity and local dimensionality reduction, thus combining both FA and Mixture model's merits which is particularly desirable for Web data.

3 MFA-Based User Segmentation

Since each mixture component models a subpopulation of Web users following the same distribution, we can naturally derive user segments based on the relationship between users and components:

$$P(g|\mathbf{u}) \propto P(g)p(\mathbf{u}|g) = P(g)\mathcal{N}_p(\mu_g, \mathbf{L}_g\mathbf{L}_g^T + \mathbf{\Psi}). \tag{6}$$

In our approach, for each of the m mixture components, we choose those users whose posterior memberships are greater than a certain threshold as its representative users. This is, essentially, a soft clustering of user sessions based on membership probabilities given in Equation 6. Hard clustering can also be achieved by simply allocating each user into only one of the segments which satisfies $argmax_g P(g|\mathbf{u})$. Such a set of x users in a segment, $U^g = \{\mathbf{u}_1^g, \mathbf{u_2}^g, \cdots, \mathbf{u}_x^g\}$, would have similar preference patterns (determined by the shared loading matrix $\mathbf{L}_g$).

In addition to user segment derivation, expected preference values for each user segment can be further derived according to Equation (7). That is, the conditional expectations of a user's preference values is obtained as the factor scores associated with a certain mixture component:

$$\begin{aligned} E(\mathbf{z}, g|\mathbf{u}) &= \int_{-\infty}^{+\infty} \mathbf{z}p(\mathbf{z}, g|\mathbf{u})d\mathbf{z} = P(g|\mathbf{u}) \int_{-\infty}^{+\infty} \mathbf{z}p(\mathbf{z}|\mathbf{u}, g)d\mathbf{z} \\ &= P(g|\mathbf{u})E[\mathbf{z}|\mathbf{u}, g] \end{aligned} \tag{7}$$

where $E[\mathbf{z}|\mathbf{u}, g] = \mathbf{L}_j^T(\mathbf{L}_j\mathbf{L}_j^T + \mathbf{\Psi})^{-1}(\mathbf{u} - \mu_j)$, and $P(g|\mathbf{u})$ is in Equation (6). Based on these preference values we can easily identify the segment's dominant factors which characterize the behavior of users within that segment.

To create an aggregate representation of the user segment U^g, we compute the weighted centroid of all the observation vectors in the segment (weighted by the membership), which results in a representation of the segment as a set of page-weight pairs. The algorithm for generating the aggregate representation of each user segment is based on the following two steps:

1. For each mixture component g, choose those user sessions with posterior memberships $argmax_g P(g|\mathbf{u})$ to form a candidate session set U^g, where $P(g|\mathbf{u})$ is determined by the posterior probability as in Equation 6.
2. Recall that each user session $\mathbf{u}_i \in U^g$ is a p-variate page vector. For each set U^g, compute its weighted centroid vector of pages as

$$\mathbf{v}^g = \frac{1}{|U^g|} \sum_{\mathbf{u}_i \in U^g} [\mathbf{u}_i \times P(g|\mathbf{u}_i)],$$

   where $|U^g|$ denotes the total number of sessions in set U^g.

Therefor for each user segment g, we derive a centroid-based user model represented as a page vector $\mathbf{v}^g$.

4 Experiments and Evaluation

In this section we evaluate the MFA-based model fit and the predictive effectiveness of our derived segments using two different data sets: CTI data and ACR

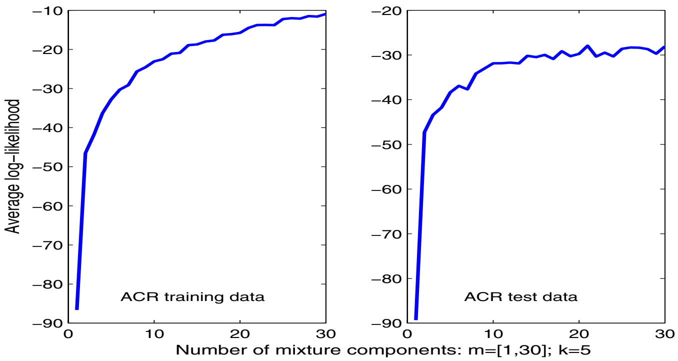


Fig. 1. MFA Log-likelihood vs. number of mixture components in ACR data

data. CTI data set is based on the server logs of the host Computer Science department spanning a one-month period containing 21,299 user sessions and 692 Web pages after preprocessing. The site is highly dynamic, involving numerous online applications, advising, faculty-specific Intranet applications, etc. Thus, we expect the discovered usage patterns to reflect various functional tasks performed by diverse groups of users. The ACR data set is based on the Web server log files of Association for Consumer Research Web site which contained 7,834 user sessions and 40 Web pages after preprocessing. Each data set was randomly divided into multiple training and test sets for the purpose of cross-validation.

4.1 Evaluation of Model Fit with Average Log-Likelihood

Average log-likelihood on training data measures the goodness of fit of the model to the training data, i.e., the likelihood that an observed case in the data is generated by the model (represented by its parameter set Θ). Specifically, the average log-likelihood $\bar{\mathcal{L}}(\Theta|data) = \frac{1}{n} \ln \prod_{i=1}^{n} p(\mathbf{u}_i|\Theta)$ of MFA is computed as follows.

$$\bar{\mathcal{L}} = \frac{1}{n} \sum_{i=1}^{n} \ln \sum_{g=1}^{m} P(g) \mathcal{N}_p(\mathbf{u}_i | \mu_g, \mathbf{L}_g \mathbf{L}_g^T + \mathbf{\Psi}); \qquad (8)$$

We should expect that the log-likelihood on the evaluation data is consistent with the training data, in order to obtain a relatively generalizable model. An example on ACR data is illustrated in Figure 1. First, we fit MFA to the training data, and vary the number of mixture components from $m \in [1, 30]$. Note that $m = 1$ corresponds to the standard single Factor Analysis. Then we evaluate the learned models on the evaluation data set based on their average log-likelihoods. Each likelihood value is an average of 5 runs of MFA EM algorithm to offset the random model initialization effect. From the training data likelihood (left) we see a significant improvement of the average log-likelihood from single FA

($m = 1$) to a mixture of two factor analyzers ($m = 2$) and a slower rate of increase thereafter. This tells us that the usage data can be better modeled by MFA than FA to address multimodal distribution problems.

Furthermore, from the test set likelihood curve (right) we can roughly tell when the model starts to over-fit. In this particular case, we see the average log-likelihood curve starts to level off with the number of mixture components $m = 10$. We found that this kind of cross-validation is effective for selecting a desirable number of mixture components although other methods such as BIC may also be applied.

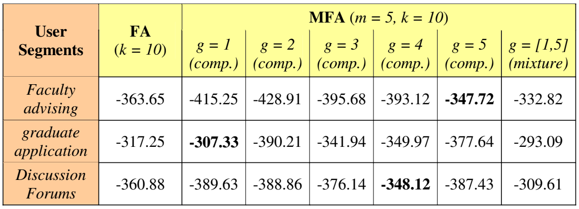
4.2 Analysis of User Segment Behavior

We have conducted another set of experiments specifically to verify the following hypotheses:

- **Hypothesis 1:** Each distribution component of the mixture model MFA corresponds to a finer-grained representation of a group of users sharing similar navigation behavior pattern than an ordinary FA does since mixture models can capture mixed underlying distributions behind the data;
- **Hypothesis 2:** Each particular component of MFA reflects an activity type of a group of users with common interests, which are represented by the corresponding dominant latent factors.

In the CTI data, we manually identified three significant activity types, namely “faculty advising”,“graduate application” and “discussion forums”. For each activity type, we isolated its corresponding user sessions resulting in three separate data sets as evaluation data. As shown in Figure 2, for factor analysis, the average log-likelihoods on the three evaluation sets are all smaller than MFA (column *mixture*). We have also found that MFA has higher likelihood value in the training data. This is not surprising since based on our experience, general Web usage data present multimodal characteristics with mixed underlying distributions.

In order to verify our hypotheses, we carried out further experiments by separately evaluating each of the individual mixture component models in MFA. As we know that each mixture component g of MFA has a corresponding parameter set. If at least one of these components can capture a distinct user behavior type better than FA, we should expect a higher likelihood when evaluated on that behavior type data than FA which models the entire training data with mixed types. Thus we applied these individual component models to all three evaluation sets to obtain average log-likelihoods respectively as shown in Figure 2 (columns indicated by *comp.*). We can see that in the segment of “faculty advising”, the component $g = 5$ has the highest likelihood (-347.72) among the five individual component models and also higher than FA (-363.65). This kind of observation is consistent on all three evaluation data sets including “graduate application” and “discussion forums” data. We further observe that although the best single component with the highest likelihood better captures a distinct



User Segments	FA (*k* = *10*)	**MFA** (*m* = *5, k* = *10*)					
		g = *1* (*comp.*)	*g* = *2* (*comp.*)	*g* = *3* (*comp.*)	*g* = *4* (*comp.*)	*g* = *5* (*comp.*)	*g* = *[1,5]* (*mixture*)
Faculty advising	-363.65	-415.25	-428.91	-395.68	-393.12	**-347.72**	-332.82
graduate application	-317.25	**-307.33**	-390.21	-341.94	-349.97	-377.64	-293.09
Discussion Forums	-360.88	-389.63	-388.86	-376.14	**-348.12**	-387.43	-309.61

Fig. 2. Likelihood comparison of FA, MFA and its individual component models

behavior type than standard FA, the combined mixture model MFA, which incorporates all the components enjoys the highest likelihoods than either FA or individual components on all the evaluation data sets. Intuitively, this is because Web user segments generally represent diversified activities while having similar dominant navigation interests, which is better captured by a combination of mixture model and factor analysis model.

It would be interesting to take a closer look at the internal structure of a mixture component model. Since each component is essentially a factor model, we want to verify whether the dominant factors in a group of users assigned to component g would match their dominant activity type. For example, from Figure 2 we know that for the evaluation data of user segment "graduate application", the best matched mixture component model is $g = 1$ ($\bar{\mathcal{L}} = -317.25$). In other words, users having been assigned to component $g = 1$ should have the dominant interest of "graduate application". In order to verify this, we selected user sessions from the training set whose membership probability equals to $\arg\max_g P[g|\mathbf{u}, g]$ and computed their mean preference scores on five latent factors. We found that the dominant factor of the segment associated with $g = 1$, which is "graduate application" as we have known, does have the largest mean factor score $avg(E[z|\mathbf{u}, g])$ as stated in hypothesis 2.

Note that in general, there exist several empirical methods to evaluate the effect of different number of mixture components and latent dimensions based on different criteria, such as eigen scree-plot and likelihood cross-validation as shown in Section 4.1. As a matter of fact, in our case, different numbers change the overall likelihood scale for both models but not their relative comparison results. Since our main purpose here is the verification of the hypotheses we are interest in, we have kept a reasonably small number, which is convenient to manage without loss of generality.

4.3 Evaluation of Segment Quality

To asses the quality of the discovered segments we use the metric *Weighted Average Visit Percentage* (WAVP) [8]. WAVP allows us to evaluate each segment based on the likelihood that a user who visits any page in the centroid profile, will visit the rest of the pages in that profile during the same session. Specifically, let T be the set of transactions in the evaluation set, and for a certain segment profile in the form of a page vector $\boldsymbol{v}^g$, let T^g denote a subset of T whose sessions

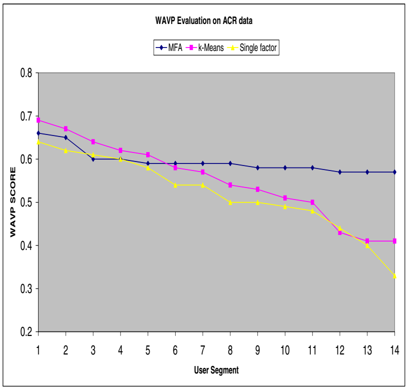


Fig. 3. WAVP evaluation on ACR data

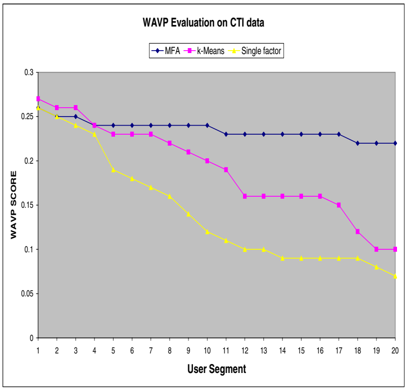


Fig. 4. WAVP evaluation on CTI data

should contain at least one page from the profile. The weighted average similarity to the profile v^g over all sessions in T^g is computed, and this average is then divided by the total weight of page in the profile:

$$WAVP\left(\boldsymbol{v}^g, T\right) = \frac{\sum_{t \in T^g} \boldsymbol{t} \cdot \boldsymbol{v}^g / |T^g|}{\sum_d weight\left(d, \boldsymbol{v}^g\right)},$$

where $weight\left(d, v^g\right)$ is the weight of a page d in this profile $\boldsymbol{v}^g$. The higher WAVP value the better the profile is, in the sense that the corresponding segment is more representative about the similar user navigational activities.

Figures 3 and 4 show the WAVP evaluation for the two data sets, comparing the MFA-based approach to FA and to standard segmentation approach using k-means clustering. All segments are ranked in descending order of WAVP. The results show that the MFA-based user models have consistently higher WAVP scores in general. Also, since MFA-based segments will generally capture more complex patterns capturing multiple factors influencing user's online behavior, the variation of WAVP scores across all MFA-based segments are significantly smaller then those of k-Means and single factor models.

5 Conclusions

The generative modeling based on FA and MFA for Web users' navigation behavior is intuitively reasonable. We can assume that users' navigation data are generated according to some distribution that is conditioned on users' hidden preferences, which can be modeled as hidden variables in latent variable models. In this paper, We have introduced an MFA-based usage mining approaches that can discover both quantitative relationship between users' manifest observations and latent factors, as well as mixture components representing user segments. Our experimental results show that our approach can successfully discover heterogeneous user segments and characterize these segments with respect of their common preferences. The aggregate representation of Web user segments, combining both of user's navigation data and the user-component memberships, can be used for explorative analysis purposes or for dynamically predicting a new user's navigational interests and recommending relevant pages or products accordingly.

References

1. R. Cooley, B. Mobasher, and J. Srivastava. Data preparation for mining world wide web browsing patterns. *Journal of Knowledge and Information Systems*, 1(1), 1999.
2. B.J. Frey, A. Colmenarez, and T.S. Huang. Mixtures of local linear subspaces for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Los Alamitos, June 1998.
3. Z. Ghahramani and G. Hinton. The EM algorithm for mixture of factor analyzers. Technical report CRG-TR-96-1, University of Toronto, 1996.
4. W. Lin, S. A. Alvarez, and C. Ruiz. Efficient adaptive-support association rule mining for recommender systems. *Data Mining and Knowledge Discovery*, 6:83–105, 2002.
5. G. McLachlan and D. Peel. Mixtures of factor analyzers. In *Proceedings of the Seventeenth International Conference on Machine Learning*, San Francisco, USA, 2000.
6. B. Mobasher. Web usage mining and personalization. In Munindar P. Singh, editor, *In Practical Handbook of Internet Computing*. CRC Press, 2005.
7. B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Effective personalization based on association rule discovery from web usage data. In *Proceedings of the 3rd ACM Workshop on Web Information and Data Management (WIDM01)*, Atlanta, Georgia, November 2001.
8. B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Discovery and evaluation of aggregate usage profiles for web personalization. *Data Mining and Knowledge Discovery*, 6:61–82, 2002.
9. L.K. Saul and M.G. Rahim. Modeling acoustic correlations by factor analysis. In M. I. Jordan and M. S. Kearn sand S. A. Solla, editors, *Advances in Neural Information Processing Systems 10*, pages 749–756. MIT Press, 1998.
10. J. Srivastava, R. Cooley, M. Deshpande, and P. Tan. Web usage mining: Discovery and applications of usage patterns from web data. *SIGKDD Explorations*, 1(2):12–23, 2000.
11. M. Wedel and W. Kamakura. *Market Segmentation: Conceptual and Methodological Foundations*. Springer, 1999.
12. Y. Zhou, X. Jin, and B. Mobasher. A recommendation model based on latent principal factors in web navigation data. In *Proceedings of the 3rd International Workshop on Web Dynamics at WWW2004*, New York, May 2004.

A Hybrid Similarity Concept for Browsing Semi-structured Product Items

Markus Zanker, Sergiu Gordea, Markus Jessenitschnig, and Michael Schnabl

University Klagenfurt
9020 Klagenfurt, Austria
markus.zanker@uni-klu.ac.at

Abstract. Personalization, information filtering and recommendation are key techniques helping online-customers to orientate themselves in e-commerce environments. Similarity is an important underlying concept for the above techniques. Depending on the representation mechanism of information items different similarity approaches have been established in the fields of information retrieval and case-based reasoning. However, many times product descriptions consist of both, structured attribute value pairs and free-text descriptions. Therefore, we present a hybrid similarity approach from information retrieval and case-based recommendation systems and enrich it with additional knowledge-based concepts like threshold values and explanations. Furthermore, we implemented our hybrid similarity concept in a service component and give evaluation results for the e-tourism domain.

1 Introduction

Browsing assistance, recommendation and personalization are key characteristics of e-commerce sites to reduce the information overload of users. Implementations of such sales support functionalities base fundamentally on the concept of similarity. Instance-based browsing allows users to retrieve similar items or items that are in principle comparable to a reference instance but should be different with respect to some specific features [1,2,3], i.e. tweaking & critiquing.

Recommendation applications differentiate themselves by the amount and form of knowledge they require about the product domain and the user situation [4,5,6]. Pure collaborative filtering approaches require no knowledge about the product domain at all. Items are solely described by an unique identifier, while extensive preference information represented by user/item ratings is required. Recommendations for a specific user are computed by determining his neighborhood of other users based on similar ratings and those items are proposed that his nearest neighbors liked. Contrastingly, content-based approaches build on data-intensive product representations, such as full-text documents or Web pages [7]. Characterizing terms from preferred document instances are used to build a user model. The further retrieval is guided by the similarity between the user model and candidate instances. Case-based [8,9] and knowledge-based [10,11] recommender systems assume structured product representations such as

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 21–30, 2006.

attribute-value pairs. Both approaches support value elicitation for user interaction. In case-based recommendation the product item most similar to the ideal case according to the explicit input of user preferences is presented. Whilst, knowledge-based recommenders possess explicit domain knowledge that maps subjective user requirements onto technical item characteristics[11]. This deep domain knowledge allows to propose product instances due to some form of business rules or constraints.

However, in application domains like e-tourism, consumer electronics or real estates such a dependency between recommendation technique and item representation is too restrictive. Typically items are semi-structured as they are represented by features as well as full-text descriptions, e.g. facilities and descriptions of a hotel or technical features and free text information about accessories of a digital camera model.

Therefore, we propose an hybrid similarity measure that encompasses similarity measures from case-based recommendation and content-based document filtering to support semi-structured item representations. Furthermore, we employ positive and negative preferences for full-text retrieval and enrich the approach with techniques from knowledge-based advisory. We add threshold values on the level of feature-similarity to explicitly model non-similarity and add explanatory facilities.

For practical applicability we developed an editor environment that ensures easy setup and maintainability. We choose the domain of e-tourism for demonstration and conducted several experimental evaluations. The paper is organized as follows: First we discuss different similarity concepts and related work in Section 2. In Section 3 we present our hybrid similarity measure. We continue by sketching our prototype and present results from an experimental evaluation in the domain of e-tourism. Finally we conclude and give an outlook on future work.

2 Related Work

Case-Based Recommendation [8] is a similarity-based retrieval of objects based on user preferences and product descriptions. There exists a large body of research on how to define similarity between entities and how it can be measured. Osborne and Bridge [12,13] distinguished similarity metrics according to their return type. When computing the similarity between a reference or query case and a set of cases, absolute or nominal similarity measures return a symbol for each case from an alphabet such as boolean values. Relative or ordinal similarity measures produce a lattice that represents a partial ordering of cases. Typically, several atomic similarity measures on different features can be combined by specific composition metrics [13,14]. However, defining knowledgeable composition operators is an effort-intensive task.

Cardinal similarity measures produce numeric similarity values that at first sight can be combined more easily by standard arithmetic operators like +, − or ×. But numeric similarity values need to be chosen carefully to ensure a correct interpretation. Nevertheless, case-based recommendation systems [8] rely on weighted sum approaches for cardinal similarity metrics as their setup and definition resembles widely known multi-attribute utility schemes.

Tweaking & critiquing is a browsing assistance mechanism that like case- and knowledge-based recommender systems also builds on structured item representations [2,15]. Users may formulate a critique concerning some of the feature values of a specific product instance c and the system returns several other product instances that fulfill the critique but are also similar to c. Newer work in this field proposes that returned items should be in addition very dissimilar towards each other [16]. Stahl [17] enhances the utility-oriented matching of a similarity-based recommendation strategy by learning from past cases. The Wasabi Personal Shopper [18] experimented with full-text wine descriptions and extracted explicit features such as *sweetness* or *tastiness*, but similarity itself is still computed on the structured case representation.

Text analysis and automatic feature selection is an important issue in the information retrieval community. There, many types of feature selection algorithms implying different computing complexities were proposed [19], e.g. based on stop words lists, minimum frequency, stemming and/or latent semantic indexing. **Content-based filtering** systems employ these techniques and learn the information seeking behavior of users by building models from preferred documents [7]. They represent documents, i.e. full-text information items, as vectors of stemmed terms. The term frequency - inverse document frequency (TFIDF) algorithm [20] for instance counts the occurrences of a term within a document and divides it by the frequency of the term throughout all indexed documents. This ensures that term occurrences with high discriminating power among all documents are reinforced. Furthermore, frequency numbers are typically normalized to eliminate the influence of document length. Although this information retrieval technique on natural language texts is quite simplistic it yields surprisingly good results compared with more complex natural language representations [21]. One of the basic assumption underlying TFIDF algorithms is that term occurrences are uncorrelated. Therefore, improvements of the algorithms take the neighborhood of terms into account and use probabilistic reasoning [22].

3 Hybrid Similarity Approach

Products and services offered over the Web are many times represented in a semi-structured form, therefore the computation of item similarity needs to take their full-text descriptions as well as their structured features into account. Here, we present a generic framework for the definition, maintenance and computation of similarity values between items that encompasses both technical approaches case-based similarity as well as document retrieval techniques from content-based filtering. Furthermore, we complement the approach with concepts from knowledge-based systems such as domain dependent threshold definitions and explanatory hints. We start by giving a small example.

3.1 Example

Our example is from the domain of accommodations, where sim_{desc} is a content-based measure for full-text descriptions while sim_{loc} and sim_{price} are functions on location and price similarity respectively.

case	description	location	price
c_1	Our farm is the ideal place for families and people looking for peace and `recreation` as well as for hikers and biking enthusiasts. We are located on a hilltop.	out in the open	28
c_2	Our farm provides `unforgettable` views on the lake and the mountains. Furthermore, we offer `organic` `produce` and relaxing `wellness` (`sauna`, solarium).	out in the open; by a mountain; near golf course; near airport	40
c_3	Spend `unforgettable` days in our emperor villa in the outskirts of Vienna. You may want to discover the city or enjoy our `wellness` and `recreational` facilities like `sauna`, jacuzzi or gym. Our restaurant serves traditional and international food as well as `organic` farm `produce`.	near town center; near golf course; near airport	85

For computing textual similarities we highlighted keywords in `typewriter` font. Note that words appearing only in a single description, e.g. jacuzzi, and those appearing in all descriptions, e.g. farm, are of no use for computing document vectors. Therefore, we get the following vector space:

keyword	c_1	c_2	c_3
recreation	1	0	1
unforgettable	0	1	1
organic	0	1	1
produce	0	1	1
wellness	0	1	1
sauna	0	1	1

We compute sim_{desc} using cosine between document vectors as given in [20]. We define similarity of location sim_{loc} using Dice coefficient [23], where twice the number of similar entries in both cases is divided by the sum of entries in each case: $sim_{loc}(c_i, c_j) = \frac{2 \times |c_i \cap c_j|}{|c_i| + |c_j|}$

Furthermore, we choose a symmetric function for sim_{price} that relates the distance of prices to the average price of both cases, i.e.

$sim_s(c_i, c_j) = 1 - \frac{2 \times \text{abs}(c_i - c_j)}{(c_i + c_j)}$

The resulting similarities for sim_{desc}, sim_{loc} and sim_{price} are therefore as follows:

$$\begin{array}{lll}
sim_{desc}(c_1, c_2) = 0 & sim_{loc}(c_1, c_2) = 0.4 & sim_{price}(c_1, c_2) = 0.65 \\
sim_{desc}(c_1, c_3) = 0.38 & sim_{loc}(c_1, c_3) = 0 & sim_{price}(c_1, c_3) = 0 \\
sim_{desc}(c_2, c_3) = 0.91 & sim_{loc}(c_2, c_3) = 0.57 & sim_{price}(c_2, c_3) = 0.28
\end{array}$$

When we now query for a similar case for c_2 we will retrieve either c_3 or c_1 as the closest one, depending on the weighting scheme for combining the atomic similarities for the three features.

3.2 Knowledge-Based Similarity

As a first step we generalize the similarity definition for case-based recommendation given in [8] by defining the similarity between a query item q and a case c as the weighted sum of m similarity functions $sim_{d_j}(q_{d_j}, c_{d_j})$, where q_{d_j} resp. c_{d_j} are feature sets of q and c. Content-based similarity computations are integrated as specific similarity functions on full-text attributes like sketched in the example.

$$sim(q,c) = \frac{\sum_{j=1}^{m} w_j \times sim_{d_j}(q_{d_j}, c_{d_j})}{\sum_{j=1}^{m} w_j} \tag{1}$$

Thus, abstracting from similarity functions on the feature level to similarity functions at the level of feature sets enhances the expressivity of similarity definitions. For instance multi-valued similarity functions can be computed on flat data structures, e.g. boolean attributes that each encode the availability of a single facility, or text indexing functions can work on combined sets of textual descriptions.

In addition we introduce a **threshold value** for each similarity definition that can be understood as a knock-out criteria. This way domain experts are enabled to explicitly exclude cases from being considered similar to some query. For instance if cases whose price differs more than 50% from the compared price should be considered dissimilar the threshold value needs to be set to 1/3.

Formally, we define a knowledge-based similarity between case c and query q as follows:

$$SimKB(q,c) = \begin{cases} 0 & : sim(q,c) < t \\ 0 & : \bigvee_{j=1}^{m}(sim_{d_j}(q_{d_j}, c_{d_j}) < t_{d_j}) \\ Sim(q,c) & : \text{else} \end{cases} \tag{2}$$

If the computed similarity at any feature dimension violates threshold t_{d_j} or the overall similarity falls below threshold t the knowledge-based similarity between q and c is set to zero. This way, domain experts can explicitly define non-similarity and exclude unintended results.

Explanations are another concept from knowledge-based approaches that we add to our similarity framework. Acceptance of a system increases and users develop more trust if they are explained why an item is proposed to them.

For each feature dimension d_j, i.e. similarity function, two explanation texts pex_{d_j} and nex_{d_j} can be formulated by the domain expert. The positively formulated one applies if the computed similarity value reaches a specific positive threshold $t_{pex_{d_j}}$. Consequently, the negatively formulated explanation text applies only in case the similarity on dimension d_j falls below threshold value $t_{nex_{d_j}}$.

$$PEx(q,c) = \{pex_{d_j} | sim_{d_j}(q,c) \geq t_{pex_{d_j}}\}$$
$$NEx(q,c) = \{nex_{d_j} | sim_{d_j}(q,c) \leq t_{nex_{d_j}}\}$$

Therefore, the similarity between query item q and c is explained by the union of all positively and negatively formulated explanation texts that apply. If the overall similarity is computed to be zero, then no explanations need to be given.

$$Ex_{SimKB}(q,c) = \begin{cases} \emptyset : SimKB(q,c) = 0 \\ PEx(q,c) \cup NEx(q,c) : \text{else} \end{cases} \tag{3}$$

Later on, a user interface component can provide an explanation button for each proposed item that delivers explanations like *The River Inn offers facilities comparable to the Oceanside Hotel. Prices are about the same range. However, it is not that close to the beach.* Placeholders allow more lively formulations such as referring to the name of the proposed or the query item. Furthermore, the domain expert has the possibility to define an order on the different explanatory strings to make sure that they are intuitively combined.

3.3 Text Similarity

In classic content-based filtering applications, where webpages or documents are recommended, similarity between items is in most cases based on preferred occurrences of keywords, i.e. if a user likes some documents, other documents are proposed that contain characteristic terms from his/her preferred documents. In application domains like email or webpage filtering a model of negative preferences is built from characterizing terms to create spam filters [24].

In the domain of product and service recommendation we propose to use both positive and negative user preferences to determine the similarity between full-text descriptions.

Descriptions of products and services must be seen more technical than news documents or Web pages, as they are typically using a limited set of domain specific terms. They deliver consistent and precise descriptions of the product or service that catch the attention of users within stringent space limitations ($\sim$ a few hundred characters). Furthermore, the assumption that what is not described is also not available holds empirically most of the time. I.e. a hotel that is situated close to a lake or offers wellness facilities will not conceal these facts to its potential customers. Therefore, if a term occurs in a document that is missing in the reference document a negative weight, i.e. a negative preference is applied.

4 Implementation

We implemented the presented techniques in a service component for B2C applications, which means it is a technical module that offers a Java API. It can be used by a shop software to realize a *show me similar items* type of functionality. Furthermore, we can employ the similarity framework to build more sophisticated recommendation services on top of it, e.g. tweaking critiquing applications or to build a user model from item characteristics preferred by the user in the past. We give an architectural overview, discuss the implementation of additional similarity measures and finally sketch the editor environment, that ensures a comfortable way of maintenance.

The system‘s overall design is based on an extensible component based architecture using the Hivemind[1] open source framework for component registration. It is part of a to-be-commercialized bunch of service components for developing interactive selling applications. Given a query item, the API returns similar items upon request that are in addition justified by an explanatory expression.

Following the generic framework approach, the system can be easily extended with additional similarity functions as long as their implementation follows the interface definition. The analysis of full-text descriptions is based on the open-source indexing engine Lucene, an open source project implementing a high performance text indexing and search engine [2]. It is the most commonly used open source IR library with a very active user and developer community [25].

For reasons of runtime efficiency, similarities between cases are pre-computed on a regular basis. At runtime, values are only retrieved from the database and no computation needs to take place. Weights between similarity functions are dynamically adapted based on user preferences. Therefore also similarities between feature sets, i.e. the results of the different similarity functions, can be stored.

Each similarity function implementation applies a scale of evaluated data, i.e. nominal, ordinal, numeric or full-text. Similarity between nominally and ordinal scaled features must be modeled explicitly by a domain expert, i.e. he/she can enter appropriate similarities into a matrix of possible feature values. For numeric features several functions, e.g. fuzzy measures or step functions are implemented. Full-text similarity values are computed using positive and negative preferences.

Knowledge acquisition and maintenance of a knowledge-based system are crucial for its effective use. Our similarity framework is part of a comprehensive suite for interactive selling applications. Therefore, the definition and configuration of similarity functions is fully supported by an editor environment based on the Eclipse Rich Client Platform. The domain expert can graphically select a set of features of a data object and associate them with a similarity function. Furthermore, threshold values and explanatory texts can be edited and a default weighting scheme defined. Within a separate testing screen, the domain expert can choose a case and query for similar items and analyze the result for plausibility.

5 Evaluation

Although a hybrid item similarity concept is advantageous from the point of expressivity alone, we also empirically evaluated two of the presented concepts

[1] See http://jakarta.apache.org/hivemind for reference

[2] See http://lucene.apache.org

by instrumenting experiments with data from the tourism domain. The data set consisted of 2127 hotels, guesthouses, farms and B&B accommodation opportunities.

In the **first experiment**, we compared the effectiveness of using positive and negative preferences for the computation of textual similarities between hotel descriptions. In order to evaluate the effectiveness of positive and negative preferences in text similarity, we created an online questionnaire with one random query document D_q and three reference documents $D_1, \ldots, D_3$ containing the full-text description of an accommodation. Each of the three reference items was computed using a different algorithm:

- positive and negative preferences $alg_{pos/neg}$, where terms occurring in both documents were multiplied with a standard factor 1 and terms that did not occur in the query document were punished with a weighting factor of -0.3.
- only positive preferences alg_{pos},
- random selection algorithm alg_{random} within the same accommodation category.

63 persons participated in the experiment and had to rate which of the three reference items in their opinion matched best with the query item. We also stored timestamp information with each submitted answer, so we could eliminate answers that did not allow themselves time for reading the accommodation descriptions. So we considered the answers of 55 participants for final evaluation: Around 51% of all answers indicated that the document retrieved by $alg_{pos/neg}$ matches the query document best. 35% rated the resulting documents from alg_{pos} highest and remaining 14% supported alg_{random}. Due to the nature of the experiment, i.e. the collection of the subjective opinion of the users, the preference random selection algorithm can be understood. Concluding, $alg_{pos/neg}$ was recommending items that had a more concise textual description of approximatively the same size as the query item while alg_{pos} has a tendency for longer documents that have a higher chance of containing most of the queried keywords. Although the experiment has a rather small sample size it nevertheless suggests that $alg_{pos/neg}$ significantly improves the retrieval results for product and service descriptions.

In the **second experiment**, we measured the performance of a hybrid similarity concept vs. pure content-based text similarity and structured item similarity.

The online questionnaire contained a semi-structured query item and again three reference items, that where computed according to the following three algorithms:

- hybrid similarity definition alg_{hybrid}, where text similarity was weighted 30% and all structured features like category, price or facilities 70%
- alone text similarity with positive and negative preferences $alg_{pos/neg}$
- pure similarity of structured features alg_{struct}.

This time, the 75 participants had to rank the three alternatives. The results of the second experiment confirmed our hypothesis that a hybrid similarity concept matches best to the concept of similarity users have in 50% of all answers rated the proposals of alg_{hybrid} best, while 18% preferred $alg_{pos/neg}$ and 31% supported alg_{struct}.

6 Conclusions

Computing the similarity between items is an essential functionality for recommendation applications in e-commerce, either to show similar items to users upon their explicit request or as an underlying capability for implementing tweaking critiquing systems as well as for building up user models. In many domains (e.g. e-tourism, consumer electronics or real estate) the offered products and services are represented by semi-structured information, i.e. a set of features and full-text descriptions. Up to now recommender systems have been either using document retrieval techniques or computed weighted sums of similarity functions on a structured feature representation. Our contribution is a hybrid framework for computing item similarity, that encompasses existing work on case-based recommendation and content-based filtering systems. Furthermore, we innovated weighted sum measures by adding knowledge concepts like threshold values, explanations and maintenance facilities. We conducted an experimental evaluation and implemented our framework as part of a suite of services for developing interactive selling applications.

References

1. Burke, R.D., Hammond, K.J., Young, B.C.: The findme approach to assisted browsing. IEEE Expert **July/Aug.** (1997) 32–40
2. Shimazu, H.: Expert clerk: Navigating shoppers‘ buying process with the combination of asking and proposing. In: 17^{th} International Joint Conference on Artificial Intelligence (IJCAI). (2001) 1443–1448
3. McCarthy, K., Reilly, J., McGinty, L., Smyth, B.: Experiments in dynamic critiquing. In: International Conference on Intelligent User Interfaces (IUI). (2005) 175–182
4. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Analysis of recommendation algorithms for e-commerce. In: ACM Conference on e-Commerce (EC). (2000) 158–167
5. Burke, R.: Hybrid recommender systems: Survey and experiments. User Modeling and User-Adapted Interaction **12(4)** (2002) 331–370
6. Adamavicius, G., Tuzhilin, A.: Towards the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering **17(6)** (2005)
7. Balabanovic, M., Shoham, Y.: Fab: Content-based, collaborative recommendation. Communications of the ACM **40(3)** (1997) 66–72
8. O‘Sullivan, D., Smyth, B., Wilson, D.: Understanding case-based recommendation: A similarity knowledge perspective. International Journal of Artificial Intelligence Tools (2005)
9. Ricci, F., Werthner, H.: Case base querying for travel planning recommendation. Information Technology and Tourism **3** (2002) 215–266

10. Burke, R.: Knowledge-based recommender systems. Encyclopedia of Library and Information Systems **69** (2000)
11. Jannach, D.: Advisor suite - a knowledge-based sales advisory system. In: European Conference on Artificial Intelligence - ECAI 2004. (2004)
12. Osborne, H., Bridge, D.: A case base similarity framework. In: 3^{rd} European Workshop on Case Based Reasoning (EWCBR). (1996)
13. Osborne, H., Bridge, D.: Similarity metrics: A formal unification of cardinal and non-cardinal similarity measures. In: 2^{nd} International Conference on Case-based Reasoning (ICCBR). (1997)
14. Adah, S., Bonatti, P., Sapino, M., Subrahmanian, V.: A multi-similarity algebra. In: ACM SIGMOD international conference on Management of data. (1998) 402–413
15. McGinty, L., Smyth, B.: Tweaking critiquing. In: Workshop on Personalisation and Web Techniques at International Joint Conference on Artificial Intelligence (IJCAI). (2003)
16. McGinty, L., Smyth, B.: The role of diversity in conversational systems. In: 5^{th} International Conference on Case-Based Reasoning (ICCBR). (2003)
17. Stahl, A.: Combining case-based and similarity-based product recommendation. In: 6^{th} European Conference on Case-Based Reasoning (ECCBR). (2006)
18. Burke, R.D.: The wasabi personal shopper: A case-based recommender system. In: 11^{th} International Conference on Applications of Artificial Intelligence (IAAI). (1999) 844–849
19. Mladenic, D.: Text-learning and related intelligent agents: A survey. In: IEEE Intelligent Systems. Volume 14. (1999) 44–54
20. Salton, G., Buckley, C.: Weighting approaches in automatic text retrieval. Information Processing and Management **24(5)** (1988) 513–523
21. Lewis, D.D., Jones, K.S.: Natural language processing for information retrieval. Communications of the ACM **39(1)** (1996) 92–100
22. Joachims, T.: A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In: International Conference on Machine Learning (ICML). (1997)
23. Frakes, W.B., Baeza-Yates, R., eds.: Information Retrieval, Data Structure and Algorithms. Prentice Hall (1992)
24. Resnick, P., Miller, J.: Pics: Internet access controls. Communications of the ACM **39(10)** (1996) 87–93
25. Gospodnetic, O., Hatcher, E.: Lucene in Action. Manning, Greenwich, CT (2005)

A Preference-Based Recommender System

Benjamin Satzger, Markus Endres, and Werner Kießling

Institute of Computer Science
University of Augsburg
D-86159 Augsburg, Germany
{satzger, endres, kiessling}@informatik.uni-augsburg.de

Abstract. The installation of recommender systems in e-applications like online shops is common practice to offer alternative or cross-selling products to their customers. Usually collaborative filtering methods, like e.g. the Pearson correlation coefficient algorithm, are used to detect customers with a similar taste concerning some items. These customers serve as recommenders for other users. In this paper we introduce a novel approach for a recommender system that is based on user preferences, which may be mined from log data in a database system. Our notion of user preferences adopts a very powerful preference model from database systems. An evaluation of our prototype system suggests that our prediction quality can compete with the widely-used Pearson-based approach. In addition, our approach can achieve an added value, because it yields better results when there are only a few recommenders available. As a unique feature, preference-based recommender systems can deal with multi-attribute recommendations.

1 Introduction

Nowadays product recommendations play a decisive role in e-shops. Many online shops like Amazon (amazon.com), Half (half.com) or CDNow (cdnow.com) use recommender systems to offer alternative or cross-selling products to their customers. Offering reliable recommendations to these customers forms a major task in advanced personalized applications to keep them as customers and to enhance sales. These recommendation techniques are typically based on collaborative filtering algorithms that can produce personal recommendations by computing the similarity between the active user's preferences and those of other users (cf. [1]). On the basis of similar users their preferred products can act as a recommendation.

The basic mechanism behind collaborative filtering systems usually takes three steps: First, looking for users (neighbors) who share the same rating patterns with the active user (for whom the prediction is for), second, computing a similarity between the neighbors and the active user for a possible weighting (these neighbors act as recommenders) and third, using the ratings from those like-minded users to generate a prediction for the active user (see figure 1).

The arguably most critical step, the computation of the similarities between users, is normally realized by computing the similarities of their votes on items

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 31–40, 2006.

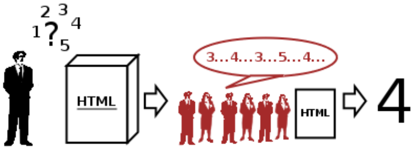


Fig. 1. The recommenders predict a rating of 4 to the active user on a *HTML*-book

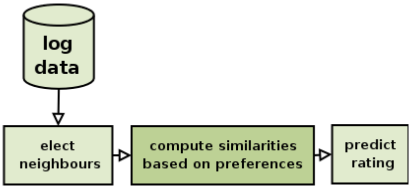


Fig. 2. Preference-based collaborative filtering

with the *Pearson correlation coefficient* [2]. The main bottleneck of all existing collaborative filtering systems is the collection of preferences (cf. [3]). To be reliable, the system needs a very large number of people (typically thousands) to express their preferences about a relatively large number of options (typically dozens). This requires quite a lot of effort from a lot of people. Since the system only becomes useful after a 'critical mass' of opinions has been collected, people will not be very motivated to express detailed preferences at the beginning stages (e.g. by scoring dozens of music records on a 10 point scale), when the system cannot yet help them. Moreover, with the Pearson correlation coefficient it is not possible to consider more than one attribute for recommendations.

In this paper we introduce a novel preference-based approach for similarity measurement, which differs from the standard collaborative filtering algorithms in the computation of the user similarity, see figure 2. In this approach we use techniques based on user preferences registered in log data on a database system. With the underlying preference model [4,5] it is also possible to use a *multi-attribute recommender*. According to [6], database driven recommender systems are rather under-researched and should get more attention by the database community. Our preference-based recommender system is a very flexible tool to support such research activities.

The rest of this paper is organized as follows: In section 2 we give an introduction to our preference model and work out the mathematical structure on similarity measurement based on user preferences. All algorithms described in this section have been implemented in our software prototype called *Recommender Framework*. The results of an evaluation of these algorithms are presented in section 3. We conclude our paper with a summary and outlook in section 4.

2 Similarity of Preferences

Now we present several approaches how to compute the similarity of two preferences. Before we focus on the similarity of preferences we give a short overview of the underlying preference model, in which preferences are modeled as *strict partial orders*. For detailed information we refer to [4].

Definition 1 (Preference). *Let* $A = \{A_1, \ldots, A_n\}$ *be a set of attribute names with corresponding domains of values* $dom(A_i)$*. The domain of* A *is defined as*

$dom(A) = dom(A_1) \times \ldots \times dom(A_n)$. Then a preference *$P$ is a strict partial order $P = (A, <_P)$, where $<_P \subseteq dom(A) \times dom(A)$.*

To facilitate the use of preferences a set of preference constructors is introduced in [4] to represent preferences in a more natural way. POS(*colors*, {red}) for example is a preference constructor to represent that *red* is preferred to all other colors in *dom*(*colors*) = {red, green, blue}.

Also [4] distinguishes between three types of preferences: *categorical base preferences*, *numerical base preferences* and *complex preferences*. In the following we briefly describe these preferences and introduce methods to compute the similarity of each type of preference.

2.1 Categorical Base Preferences

A categorical preference $P = (A, <_P)$ is characterized by a finite domain *dom(A)*. The easiest case to define a similarity is for complete preferences. Any categorical preference can be extended to a complete preference with the aid of linear extensions.

Definition 2 (Complete preference). *A categorical preference $P = (A, <_P)$ is* complete*, if $<_P$ is total, i.e. $\forall a, b \in dom(A) : a <_P b \;\vee\; b <_P a$.*

Obviously, a complete preference can be characterized as a permution of $dom(A)$. Since there exist a couple of similarity measures on the set of permutations of a set, we simply use those metrics to compute the similarity of two complete preferences. A frequently used metric in this context is *Spearman's* ρ [7] which leads to values between 0 (mismatch) and 1 (perfect match). Adapted to complete preferences P_1 and P_2 and $n = |dom(A)|$, Spearman's ρ is

$$\rho(P_1, P_2) = 1 - \frac{6\left(\sum_{a \in dom(A)} (rank_{P_1}(a) - rank_{P_2}(a))^2\right)}{n(n^2-1)}$$

where the *rank* is defined as follows:

Definition 3 (Rank). *Let $P = (A, <_P)$ be a complete preference. Then each element in $dom(A)$ can be assigned a* rank *as*

$$rank_P : dom(A) \to \mathbb{N} \backslash \{0\},\; a \mapsto |\{r \in dom(A) : r <_P a\}| + 1$$

Moreover we need the concept of linear extensions.

Definition 4 (Linear extension). *A complete preference $P_c = (A, <_{P_c})$ is a* linear extension *of a categorical preference $P = (A, <_P)$, if*

$$\forall a, b \in dom(A) : \; a <_P b \Rightarrow a <_{P_c} b$$

Example 1. Let P(color,$<_P$) be a preference with $dom(A)$={red, brown, yellow, blue} and $<_P$={(yellow, red),(yellow, brown)}. Some linear extensions of P are for example (yellow, red, brown, blue), (yellow, brown, red, blue), (blue, yellow, red, brown). But (red, yellow, brown, blue) is not a linear extension of P because the element (red, yellow) conflicts with (yellow, red) of P.

Given the notions of *complete preference* and *linear extension*, we can compute the similarity of two categorical preferences P_1 and P_2 as follows:

Algorithm 1 (Similarity of categorical preferences - Base Approach)

a) Compute all linear extensions of P_1 and P_2 (e.g. with the Varol-Rotem-algorithm [8]).

b) Compute the similarities of all linear extensions of P_1 with all linear extensions of P_2 using a metric for complete preferences (e.g. Spearman's ρ) and average the results.

This approach to compute the similarity of categorical preferences has the disadvantage that all linear extensions have to be computed. But this is often impossible because a preference can have up to $n!$ linear extensions (where n is the number of elements in $dom(A)$). Indeed [9] showed that the simpler task of counting the number of linear extensions is #P-complete. Thus this approach is only practicable for categorical preferences with small sets of attributes.

To cope with the limitations of algorithm 1 we can use a *Markov Chain Monte Carlo* (MCMC) approach [10,11]. This enables us to sample randomly from the set of linear extensions of a preference. In this vein it is possible to approximate the set of all linear extensions of a preference in a certain way. With the application of a MCMC, algorithm 1 to compute the similarity of two categorical preferences P_1 and P_2 changes as follows:

Algorithm 2 (Similarity of categorical preferences - MC-Approach)

a) Sample k linear extensions from P_1 and P_2.

b) Compute the similarities of these linear extensions analog to algorithm 1.

Using MCMC allows us to handle categorical preferences based on bigger attribute sets than algorithm 1. But the generation of one sample costs about $O(n^5\ log\ n + n^4\ log\ \epsilon^{-1})$ (depending on the used Markov chain, where ϵ is the desired accuracy and n is the number of elements in $dom(A)$).

The following algorithm to compute the similarity of two categorical preferences P_1 and P_2 is designed to deal with big attribute sets. It is much faster, but also more inexact (cf. [12]) and is based on the number of *consistent* and *inconsistent elements* of P_1 and P_2.

Definition 5 (Number of consistent and inconsistent elements). *Given two preferences $P_1 = (A, <_{P_1})$ and $P_2 = (A, <_{P_2})$. Then we define:*

a) The consistent elements *$numC(P_1, P_2)$ of P_1 and P_2 is the cardinality of elements of the set $<_{P_1} \cap <_{P_2}$.*

b) The inconsistent elements *$numI(P_1, P_2)$ is the cardinality of $<_{P_1} \cap <^{\delta}_{P_2}$, where $<^{\delta}_{P_2}$ is the dual preference which reverses the order of $<_{P_2}$.*

Example 2. Let A be an attribute with $dom(A) = \{a, b, c, d\}$ and $P_1 = (A, <_{P_1})$, $P_2 = (A, <_{P_2})$ two preferences with $a <_{P_1} b <_{P_1} c <_{P_1} d$ and $a <_{P_2} b <_{P_2} d <_{P_2} c$. Then $numC(P_1, P_2) = 5$ and $numI(P_1, P_2) = 1$ since $<_{P_1} \cap <_{P_2} = \{(a, b), (a, c), (a, d), (b, c), (b, d)\}$ and $<_{P_1} \cap <^{\delta}_{P_2} = \{(c, d)\}$.

The algorithm outlines as follows:

Algorithm 3 (Similarity of categorical preferences - CI-Approach)
a) Identify the number of consistent and inconsistent elements of the preferences P_1 and P_2
b) Compute the similarity as

$$sim(P_1, P_2) = \frac{1}{2} + \frac{numC(P_1, P_2) - numI(P_1, P_2)}{n\ (n-1)}, \quad n = |dom(A)|$$

With algorithms 1, 2 and 3 methods are available to assign a similarity to categorical preferences of different magnitudes. Now let's have a look on how to compute the similarity of numerical base preferences.

2.2 Numerical Base Preferences

Unlike categorical preferences, which are based on a finite attribute set, numerical preferences are based on a numerical domain, which can be infinite. [4] provides a number of numerical preference constructors: LOWEST, HIGHEST, AROUND, BETWEEN and SCORE. The LOWEST(price) preference could for example express that a person prefers lower values for 'price' over higher values; BETWEEN(year of construction, [1960,1970]) in the context of cars would express that a person likes cars most that were made during the 60's. The most expressive numerical preference is the SCORE preference, as the other numerical preferences can be expressed by a SCORE preference.

A SCORE preference is based on a SCORE function $f : dom(A) \to \mathbb{R}$. Then P = SCORE(A,$f$) is a SCORE preference, if

$$x <_P y \Longleftrightarrow f(x) < f(y)\ .$$

To express for instance a HIGHEST preference by a SCORE preference, choose $f(x) = x$. In this case holds: $x <_P y \Longleftrightarrow x < y$, so higher values are preferred. Table 1 shows how to choose the SCORE function of a SCORE preference to represent the particular numerical preferences.

Table 1. Choice of the SCORE function, cf. [12]

preference	choice of f
LOWEST(A)	$f(x) = -x$
HIGHEST(A)	$f(x) = x$
AROUND(A, $\overline{x}$)	$f(x) = \begin{cases} x & \text{if } x \le \overline{x} \\ 2\,\overline{x} - x & \text{if } x > \overline{x} \end{cases}$
BETWEEN(A, $[x_1, x_2]$)	$f(x) = \begin{cases} x_1 & \text{if } x \in [x_1, x_2] \\ x & \text{if } x < x_1 \\ x_1 + x_2 - x & \text{if } x > x_2 \end{cases}$

With this knowledge we can restrict the computation of the similarity of numerical preferences to SCORE preferences. So we examine the SCORE function f to compute the similarity of SCORE preferences. Because of the non-finiteness of numerical preferences, the concepts for categorical preferences can't be applied here and another method for similarity computation is necessary.

Algorithm 4 (Similarity of SCORE preferences)
Let $P_1 = SCORE(A, f_1)$ and $P_2 = SCORE(A, f_2)$ denote two SCORE preferences. The algorithm to compute their similarity $sim(P_1, P_2)$ is as follows:

a) Choose values for a and b, where $[a, b]$ should include all existing values of the attribute A (this can be done automatically).
b) Compute with an algorithm of numerical integration and differentiation:

$$sim(P_1, P_2) = 1 - min\left(\frac{\int_a^b |f_1' - f_2'|}{2 \cdot (b-a)}\ ,\ 1\right)$$

Example 3. Let P_1 = BETWEEN$(A, [5, 8])$ and P_2 = HIGHEST(A) two numerical preferences with $dom(A) = \mathbb{R}$ within $[0, 10]$. Figures 3 and 4 show the derivations of the choosen SCORE preferences f_1 and f_2 according to table 1.

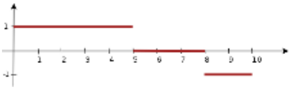


Fig. 3. f_1' in $[0, 10]$

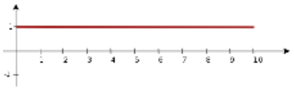


Fig. 4. f_2' in $[0, 10]$

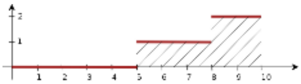


Fig. 5. $|f_1' - f_2'|$ in $[0, 10]$

The hatched area in figure 5 represents $|f_1' - f_2'|$ and for the computation of the similarity of P_1 and P_2 we integrate $|f_1' - f_2'|$ on the interval $[0, 10]$ which leads to the similarity $sim(P_1, P_2) = 1 - \frac{\int_0^{10} |f_1' - f_2'|}{2 \cdot (10-0)} = 0.65$, what is intuitively an adequate value for a BETWEEN$(A, [5, 8])$ and HIGHEST(A) comparison.

2.3 Complex Preferences

Complex preferences decide on how important certain preferences are for a person. Supposed Anne and Julia both have the preferences POS(colour, {red}) und LOWEST(price) for hats. For Anne the red color of the hat is more important than a low as possible price, whereas for Julia the price is decisive. Anne and Julia, only considered on the basis of their base preferences, are identical. A more exact classification of their similarity is possible, if *complex preferences* are involved. Because of the different weighting of the preferences of the two women, their purchasing patterns would likely differ. Thus the consideration of complex preferences, computing the similarity, is absolutely advisable.

To include the complex preferences, we build a *preference order* based on all complex preferences of a person and afterwards compare the users with these preference orders.

Definition 6 (Preference order). *Let $P = \{P_1, ..., P_n\}$ a set of preferences. A* preference order *PO is a partial order $PO = (P, \leq_{PO})$ with $\leq_{PO} \subseteq P \times P$. It represents an order of the preferences of a person.*

Due to the limited space of this paper we only consider two complex preferences, *Pareto* $'\otimes'$ (equally important) and *prioritized* $'\&'$ (more important).

If C is a set of complex preferences of a user and P_1 and P_2 are equally important we put $P_1 \leq_{PO_C} P_2 \wedge P_2 \leq_{PO_C} P_1$ in our preference order. For a prioritized preference P_1 & P_2 we consider $P_2 \leq_{PO_C} P_1$ in the preference order.

The definitions of the Pareto, prioritized and further complex preferences as well as details for the assignment of a preference order are given in [12]. Now we can outline the algorithm to compute the similarity of preference orders.

Algorithm 5 (Similarity of preference orders)
Given two users U_1 and U_2.

a) Build the preference orders PO_1 and PO_2 based on the complex preferences of U_1 respectively U_2
b) Compute the similarity $POsim(PO_1, PO_2)$ of PO_1 and PO_2 as follows:

$$\frac{1}{2} + \frac{numC(PO_1, PO_2) - numI(PO_1, PO_2) + numEq(PO_1, PO_2)}{n\ (n-1)}$$

where $numC(\cdot,\cdot)$ and $numI(\cdot,\cdot)$ are defined analog to definition 5. $numEq(\cdot,\cdot)$ is the number of equal important preferences and n is the number of possible preferences.

Finally, it is interesting how to compute the similarity of users based on their preferences.

2.4 Similarity of Users

The fundamental part of a collaborative filtering recommender system is the computation of a similarity between two users. Our preference-based approach works as follows.

Algorithm 6 (Similarity of users)
Given two users U_1 and U_2, compute their similarity as follows:

a) Based on the complex preferences C_{U_1} and C_{U_2} assign a preference order PO_{U_1} to U_1 and a preference order PO_{U_2} to U_2.
b) Compute the similarity of U_1 and U_2 as follows:

$$\frac{\omega\ POsim(PO_{U_1}, PO_{U_2})\ +\ \sum_{i=1}^{n}(\omega_i\ sim(P_{U_1,i}, P_{U_2,i}))}{\omega\ +\ \sum_{i=1}^{n}\omega_i}$$

$P_{U_1,i}$ and $P_{U_2,i}$ are base preferences of U_1 and U_2, respectively. With the weights ω and $\omega_1, \ldots, \omega_n$ one can affect the influence of the different preferences.

Example 4. Given two persons Anne and Julia and their preferences concerning cars, let's consider two attributes 'manufacturer' (M) with domain $dom(M) = \{\text{VW, BMW, Fiat}\}$ and 'cylinder capacity' (C) with values in [50, 5000].

Anne and Julia have the same base preferences P_1 = POS(M,{BMW}) and P_2 = HIGHEST(C), but they differ in the way they valuate the preferences. For Anne the attributes 'manufacturer' and 'cylinder capacity' are equally important ($P_1 \otimes P_2$) whereas Julia regards high values for cylinder capacity for more important than a BMW (P_2 & P_1).

Without looking at the complex preferences the similarity of both would be '1'. Taking the complex preferences into account and building preference orders like $P_1 \leq_{PO_A} P_2 \wedge P_2 \leq_{PO_A} P_1$ for Anne and $P_2 \leq_{PO_J} P_1$ for Julia this results in a similarity of 0.83 using algorithm 6, i.e. they have very similar preferences.

Detailed description of similarity measures on user preferences and more examples are given in [12].

3 Evaluation

The algorithms described in section 2 have been evaluated in comparison with the well established and widely used Pearson collaborative filtering algorithm. For evaluation we used our *Recommender Framework* [12], a software prototype where all these algorithms have been implemented.

We predicted the vote of a user for a movie in the data set from the well known EachMovie collaborative filtering service (www.cs.umn.edu/Research/GroupLens). This data set consists of 948 users who voted movies with 1 (worst) to 5 (best). The data set contains 100.000 votes concerning 1.682 different movies.

Since the information of the EachMovie data set is very sparse and we can use multi-attribute recommendation with our preference approach, we extended the originally EachMovie data set with further informations from the Internet Movie Database (www.imdb.com) such as director, country, language or runtime. This new data set was splitted into five different and disjoint training and test data sets. The training data consists of 80%, the test data set of 20% of the raw data.

One experiment is made up of five passes, each uses another training/test data set. Based on such a training data set all 20.000 votes of the corresponding test data will be predicted with the chosen algorithm. For elicitation of user preferences concerning the different attributes from the movie database we use the Preference Miner [13], a database tool for mining user preferences from users' log data. The Preference Miner detected a lot of preferences concerning the *MovieID* and *Director* attributes, i.e. POS preferences. We evaluated three different algorithms which only differ in the computation of the similarity of users to the active user:

- **Pearson:** The similarity was computed by the well known Pearson correlation coefficient ([2]).
- **Preference_MovieID:** Here we used the preference-based algorithm; only the attribute *MovieID* was considered.

Table 2. Results of the evaluation - MAE

Algorithm	**1**	**5**	**10**	**20**	**50**
Pearson	0.97397	0.79674	**0.76296**	**0.74621**	**0,74049**
Preference_MovieID	0.95841	0.79499	0.77095	0.75773	0,75378
Preference_MovieID+Director	**0.95442**	**0.79420**	0.76852	0.75534	0,75143

- **Preference_MovieID+Director:** Additional to *MovieID* we considered the attribute *Director* for user comparison in the preference-based case.

Also we used a different number of neighbors for all algorithms: 1, 5, 10, 20 and 50. The quality of prediction was measured with the mean average (distance) error (MAE) and averaged at the end of one experiment. The results are given in table 2 where the bold values represents the best outcomes (cf. [12]).

Observations:

- Obviously the results are getting better with increasing number of neighbors. In the situation where only a few neighbors were examined, our preference-based approach was better than Pearsons algorithm. This is very useful for recommender systems where only a few neighbors are available, e.g. in newly founded online shops.
- The multi-attribute recommender is better than the single attribute algorithm in all cases and sometimes better than Pearson. In combination with feasible attributes it could achieve improved results.

The result of this comparison shows that preference mining recommendations compare very well with the Pearson collaborative filtering method. However, the computation of the prediction is more time-consuming with increasing number of neighbors. In addition to the pretty good accuray, preference-based recommendations have some unique important advantages. The similarity of users is based on intuitively understandable preferences. Moreover, the recommendations can be based on more than one attribute whereas Pearson's method can only work with one attribute.

4 Summary and Outlook

In this paper we have introduced a mathematical framework for the similarity of preferences, adopting a familiar preference notion from database systems. We evaluated the recommendation quality of all presented algorithms with our software prototype. The results of our extended EachMovie use case give strong evidence that our preference-based similarity approach can compete with the widely-used Pearson approach. It even leads to better recommendations if there are only a few neighbours avalaible. Clearly this achieves added value for start-up e-shops with only a few customers or even for established e-businesses, where some parts of the sales items attract lesser customer attention. Moreover, user ratings need not be collected manually, but can be automatically mined from log files. As a unique feature of our preference-based approach, we are able to deal with recommendations that are based on multiple attributes.

Currently above advantages come at some price due to an increased runtime of our preference-based algorithms compared to the Pearson approach. Investigating how to reduce this overhead will be our next step towards powerful recommender systems and online preference-based recommendations, which are based on preference techniques from database systems.

Acknowledgement. We would like to thank Stefan Holland for his comments which helped to improve this paper.

References

1. J. S. Breese, D. Heckerman, and C. Kadie: *Empirical Analysis of Predictive Algorithms for Collaborative Filtering.* In Proc. of the 14th Annual Converence on Uncertainty in Artificial Intelligence, p. 43–52, Madison, Wisconsin, USA, 1998.
2. P. Resnik, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: *An open architecture for collaborative filtering of netnews.* In Proceedings of the ACM, Conference on Computer Supported Cooperative Work, p. 175–186, 1994.
3. U. Shardanand, and P. Maes: *Social information filtering: Algorithms for automating 'word of mouth'.* In Proceedings of the Conference on Human Factors in Computing Systems, p. 210–217, 1995.
4. W. Kießling: *Foundations of Preferences in Database Systems.* In Proceedings of the 28th International Conference on Very Large Databases (VLDB 2002), p. 311–322, Hong Kong, China, 2002.
5. J. Chomicki: *Preference Formulas in Relational Queries.* In the ACM Transactions on Database Systems (TODS), Volume 28(4), p. 427-466, 2003
6. Y. Ioannidis, and G. Koutrika: *Personalized Systems: Models and Methods from an IR and DB Perspective.* In the 31th International Conference on Very Large Databases (VLDB 2005), Tutorial, Trondheim, Norway, 2005.
7. C. Spearman: *The proof and measurement for association between two things.* American Journal of Psychology, 15 p. 72–101, 1904.
8. Y. Varol, and D. Rotem: *An algorithm to generate all topological sorting arrangements.* Computer Journal, 24 p. 83–84, 1981.
9. G. Brightwell, and P. Winkler: *Counting Linear Extensions.* Order, 15(3), p. 225–242, 1904.
10. A. Karzanov, and L. Khachiyan: *On the conductance of order markov chains.* Order, 8(1), p. 7–15, 1991.
11. M. Jerrum, and A. Sinclair: *The markov chain monte carlo method: an approach to approximate counting and integration.* PWS Publishing, Boston, MA, 1996.
12. B. Satzger: *Development and evaluation of a software prototype to generate preference-based recommendations* (in German). Diploma thesis, Chair for Databases and Information Systems, University of Augsburg, Dec. 2005.
13. S. Holland, M. Ester, and W. Kießling: *Preference Mining: A Novel Approach on Mining User Preferences for Personalized Applications.* In Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2003), p. 204–216, Dubrovnik, Croatia, 2003.

Enhanced Prediction Algorithm for Item-Based Collaborative Filtering Recommendation

Heung-Nam Kim[1], Ae-Ttie Ji[1], and Geun-Sik Jo[2]

[1] Intelligent E-Commerce Systems Laboratory,
Department of Computer Science & Information Engineering, Inha University
{nami, aerry13}@eslab.inha.ac.kr
[2] School of Computer Science & Engineering, Inha University,
253 Yonghyun-dong, Incheon, Korea 402-751
gsjo@inha.ac.kr

Abstract. As the Internet infrastructure has been developed, a substantial number of diverse effective applications have attempted to achieve the full potential offered by the infrastructure. Collaborative Filtering recommender system, one of the most representative systems for personalized recommendations in E-commerce on the Web, is a system assisting users in easily finding the useful information. But traditional collaborative filtering suffers some weaknesses with quality evaluation: the sparsity of the data, scalability, unreliable users. To address these issues, we have presented a novel approach to provide the enhanced prediction quality supporting the protection against the influence of malicious ratings, or unreliable users. In addition, an item-based approach is employed to overcome the sparsity and scalability problems. The proposed method combines the item confidence and item similarity, collectively called *item trust* using this value for online predictions. The experimental evaluation on *MovieLens* datasets shows that the proposed method brings significant advantages both in terms of improving the prediction quality and in dealing with malicious datasets.

1 Introduction

With the explosive growth of the Internet, recommender systems have been issued as a solution for the problem of information overload. Recommender systems intend to assist users in finding the information most relevant to their preferences [11]. One of the most successful technologies in recommender systems is Collaborative Filtering (CF) and numerous commercial systems apply this technology to serve recommendations to their customers. The traditional the task in CF is to predict the utility of a certain item for the target user (often called active user) from the user's previous preference or the opinion of other similar users, and make appropriate recommendations [2]. However, despite the success and popularity, traditional CF encounters several limitations, namely sparsity, scalability, cold start, and the malicious ratings problem. And a number of researches have been proposed and challenged to address these problems related to collaborative filtering [2, 5, 6, 7, 10, 13].

In this paper, the techniques of CF are exploited, in generating enhanced predictions derived from explicit ratings. The main objective of this research is to develop a robust approach that provides high-quality predictions and recommendations even

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 41 – 50, 2006.

when some ratings of users are unreliable. In addition, an item-based approach is employed to overcome the sparsity and scalability problems [2]. The proposed approach first determines the similarities between the items and subsequently identifies the confidence of the items, indicating the accuracy of the past predictions. Furthermore, this paper presents a method of combining the item confidence and item similarity, collectively called *item trust* using this value for online predictions and recommendations. The subsequent sections of this paper are organized as follows: The next section contains a brief overview of some related researches. In section 3, the approach for CF, based on *item trust*, is described. The performance evaluation is presented in section 4. Finally, we remark the conclusions and future works.

2 Background and Related Work

This section briefly explains previous researches related to CF-based recommender systems, which can be divided into two classes: Memory-based CF and Model-based CF [1]. Since the first system to generate automated recommendations, the GroupLens [3], was proposed, the user-based approach has been the most widely used for recommendation systems. User-based CF uses a similarity measurement between neighbors and the target users to learn and predict the preference towards new items or unrated products regarding a target user. Though user-based CF algorithms tend to produce more accurate recommendations, they have some serious problems relating to the complexity of computing each recommendation as the number of users and items grow. In order to improve scalability and real-time performance in large applications, a variety of model-based recommendation techniques were developed [2, 3, 12]. Especially, a new class of Item-based CF, which is one of model-based approaches and this research focuses on, has been proposed. This approach provides item recommendations by first developing a model of user ratings. In comparison to user-based approaches, item-based CF is typically faster in terms of recommendation time, though the method may have an expensive learning or model building process [4]. Instead of computing the similarities between the users, item-based CF reviews a set of items the target user has rated and selects k most similar items, based on the similarities between the items. Sarwar et al. [2] evaluated various methods to compute similarity and approaches to limit the set of item-to-item similarities that must be considered. And Deshpande et al. [5] proposed Item-based top-N recommendation algorithms that are similar to previous item-based schemes. They separated the algorithms two distinct parts for building a model of item-to-item similarities and deriving the top-N recommendations using this pre-computed model. Despite effectiveness of item-base CF algorithms, they still have some weaknesses concerning data sparseness, cold start users and ratings of malicious users. Hence, a number of recent research efforts that focus on the use of trust concepts during the recommendation process [6, 7, 8]. In addition, distributed recommender systems have been proposed to deal with the existing weaknesses [7, 10, 12].

3 Collaborative Filtering Based on Item Trust

The proposed method is divided into two phases, an offline phase and an online phase. The offline phase is a building model phase, and the online phase is either a prediction or recommendation phase. Fig. 1 illustrates a brief overview of the system.

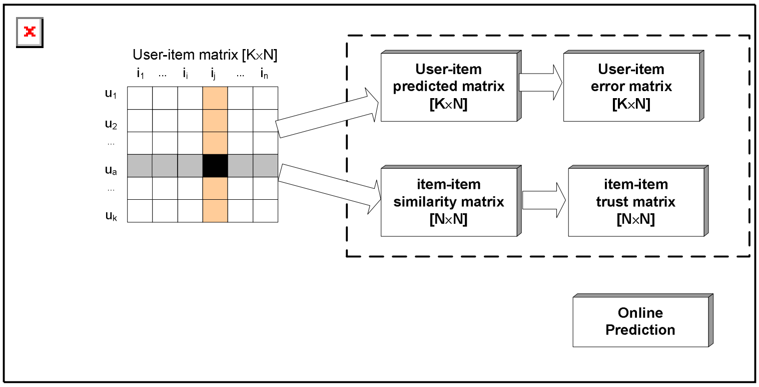

User-item matrix [K×N]
i1 ... ii ij ... in
u1
u2
...
ua
...
uk
User-item predicted matrix [K×N]
User-item error matrix [K×N]
item-item similarity matrix [N×N]
item-item trust matrix [N×N]
Online Prediction

Fig. 1. Collaborative filtering recommendation based on item trust: item-based approach

3.1 Cosine-Based Similarity with Inverse Item Frequency

The most important task in CF recommendation is the similarity measurement because different measurements lead to different neighboring users or items, in turn, leading to different recommendations. From the item-based similarity viewpoint, there are several different methods of computing the similarity between items, such as correlation-based similarity, cosine similarity, and adjusted cosine similarity [2]. Initially, the methods that would be more accurate in the proposed system were examined. As a result, the cosine measures greater accuracy than the other measures (see Table 1).

In cosine similarity between items, two items are treated as two vectors in the space of users. In addition, we also consider the number of users' ratings for items as mentioned in [5]. Consider two users A and B, both of whom have co-rated item *i* and *j*, however user A rated just 5 items whereas user B rated 100 items. In this situation, user A, rating fewer items, is a relatively more reliable for the similarity of items *i* and *j* than user B rating lots of items. Therefore, the inverse user frequency as described in [1] for the proposed system is modified, namely *the inverse item frequency*.

In a system which users have co-rated items, *the inverse item frequency* can be applied to the cosine similarity technique. The similarity between two items, *i* and *j* is measured by equation (1).

$$Sim(i,j) = \frac{\sum_{u\in User} r_{u,i}\cdot r_{u,j}\cdot f_u^{\,2}}{\sqrt{\sum_{u\in User} r_{u,i}^{\,2}\cdot f_u^{\,2}}\sqrt{\sum_{u\in User} r_{u,j}^{\,2}\cdot f_u^{\,2}}} \qquad (1)$$

where *User* is a set of users who both rated *i* and *j*, $r_{u,i}$ is the rating of user *u* on item *i*, and $r_{u,j}$ is the rating of user *u* on item *j*. The inverse item frequency of user *u*, f_u is defined as $log(n/n_u)$ where n_u is the number of items rated by user *u* and *n* is the total number of items in the database. If user *u* rated all items, then the value of f_u is 0. Likewise the inverse user frequency, the main idea of the inverse item frequency is that users rating lots of items present less contribution with regard to prediction, than users rating a smaller number of items.

3.2 Item Confidence for Computing Item Trust

Before describing the algorithms, some definitions of the matrices are introduced.

−User-Item actual rating matrix. If there is a list of k users U={u_1,u_2,…,u_k}, a list n items I={i_1,i_2,…i_n}, and a mapping between user-item pairs, and the explicit ratings, $k \times n$ user-item data can be represented as a rating matrix. This matrix is called a *User-Item actual rating matrix*, *A*. The matrix rows represent users, the columns represent items, and $A_{a,j}$ represents the rating of a user a on an item *j*. Some of the entries are not filled, as there are items not rated by some users.

−User-Item predicted rating matrix. This is a matrix of users and items that have the predicted values for users on items. From a matrix *A*, the system can predict $P_{a,i}$ for a given target item *i* which has already been rated by target user *a*. This matrix is called a *User-Item predicted rating matrix*, *P*. Likewise a matrix *A*, The matrix rows represent users, the columns represent items and the elements of the matrix *P* is a subset of the elements of a matrix *A*, $P \subseteq A$.

−User-Item error matrix. From the given set of actual and predicted rating pairs <$A_{a,j}$, $P_{a,j}$> for all the data in a matrix *A* and *P*, a *User-Item error matrix*, *E*, can be represented as absolute error, which can be computed by subtracting the predicted rating for users on items from the actual rating for users on items. The elements of A matrix *E* is also a subset of the elements of a matrix *A*, $E \subseteq A$.

For constructing a matrix *E*, firstly a user's rating should be predicted for an item which has already been rated. For the purpose of this, a user-based Resnick prediction measure can be modified, which was introduced by [3], to an item-based prediction measure, as presented in equation (2). The prediction for the target user *a* on item *i*, $P_{a,i}$, is obtained as the following:

$$P_{a,i} = \overline{A_i} + \frac{\sum_{j \in N(a)} (A_{a,j} - \overline{A_j}) \cdot sim(i, j)}{\sum_{j \in N(a)} | sim(i, j) |} \tag{2}$$

where N(a) is the set of *k* most similar items which the user *a* rated and $A_{a,j}$ is the rating of the user *a* on item *j*. In addition, $\overline{A_i}$ and $\overline{A_j}$ refer to the average rating of the item *i* and *j*. *sim(i, j)* represents the similarity between the items *i* and *j*, which is calculated as mentioned in equation (1).

Once the predictions for users on items are represented on a user-item predicted rating matrix, absolute error of each prediction can be computed for constructing a user-item error matrix. Given the set of actual and predicted rating pairs <$A_{u,j}$, $P_{u,j}$> for all data in the user-item matrices, an absolute error, $E_{u,j}$, is calculated as:

$$e_{u,j} = | A_{u,j} - P_{u,j} |$$

As a result of the error matrix, the confidence of an item, indicating the percentage of accurate predictions for an item, is computed from each column in the user-item error matrix and is defined as the following equation (3).

$$confidence(j) = \sum_{u \in U} \frac{| E_{u,j} \cap E^{r}_{u,j} |}{| E_{u,j} |} \tag{3}$$

where $E_{u,j}$ is a set of errors predicted for user u on item j and $E^{r}_{u,j}$ is the set of errors for which an absolute error of $e_{u,j}$ is within a predefined ε ($e_{u,j} < \varepsilon$). *U* is the set of users rating item *j*. For example, given item *j*, if a hundred errors have been computed for an item *j* and eighty of theses predictions are accurate, the confidence of item *j*, *confidence(j)*, is 0.8.

3.3 Prediction Based on Item Trust

As mentioned previously, the item-based CF approach builds a model of item similarity, which can be achieved offline, prior to online prediction or recommendation. Since most of tasks can be conducted in the offline phase, this approach can result in fast online performance. In addition, this assists in solving the sparsity and scalability problems [2, 5]. The proposed method also provides another advantage, the ability to protect the influence of malicious ratings.

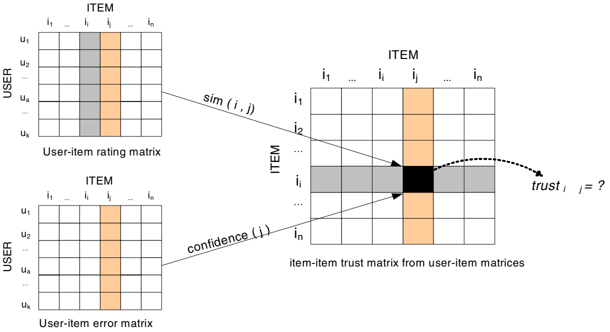


Fig. 2. The item-item matrix for a pair of items trust from the user-item matrices

In order to support fast online predictions, the trust value between two items is calculated in offline, namely *item-item trust matrix*. Fig. 2 illustrates the process of the item-item trust matrix construction from the user-item matrices.

−Item-item trust matrix. The item trust model can be represented as a matrix, *T*, in which rows and columns are both items. An entire $n \times n$ item-item trust matrix can be filled in, given by the $k \times n$ user-item matrices, *A* and *E* using equation (4)

$$trust^{\beta}{}_{i \to j} = \frac{(\beta^2 + 1) \cdot sim(i, j) \cdot confidence\ (j)}{\beta^2 \cdot sim(i, j) + confidence\ (j)} \tag{4}$$

where parameter β is specified for adjusting the relative weighting between the similarity of items and the confidence of an item. If β=0 then $trust^{\beta}{}_{i \to j}$ just takes *sim(i,j)* into account whereas if β=+∞ then $trust^{\beta}{}_{i \to j}$ just coincides with *confidence(j)*. When a value of $\beta = 1$ is used, the equal importance to *sim(i,j)* and *confidence(j)* is considered. The trust value between a pair of items is in the range of [0, 1] and is not symmetric ($trust^{\beta}_{i\rightarrow j} \neq trust^{\beta}_{j\rightarrow i}$). The appropriate value for β is selected by performing experimental analysis.

The most important task in a CF is to generate the prediction, attempting to guess the rating that a user would provide for an item [2]. In order to compute the predicted rating of target user *a* for the target item *i*, the item-based Resnick prediction measure discussed in section 3.2 is used. However, instead of using item similarity, *sim(i,j)*, the prediction algorithm in the online phase, uses the item trust value, $trust^{\beta}_{i\rightarrow j}$ as defined in equation (5).

$$P_{a,i} = \overline{A_i} + \frac{\sum_{j\in N(a)} (A_{a,j} - \overline{A_j}) \cdot trust^{\beta}_{i\rightarrow j}}{\sum_{j\in N(a)} trust^{\beta}_{i\rightarrow j}} \tag{5}$$

where N(a) is the set of *k* most similar items which the user *a* rated and $A_{a,j}$ is the rating of the user *a* on item *j*. In addition, $\overline{A_i}$ and $\overline{A_j}$ refers to the average rating of the items *i* and *j*.

4 Experimental Evaluation

In this section, experimental results of the proposed method are presented. In order to compare the performance of the proposed method, user-based and item-based CF recommendation systems were implemented. All experiments were carried out on a Pentium IV 3.0GHz with 1GB RAM, running MS-window 2003 server. In addition, the recommendation system for the web was implemented using MySQL 4.0 and PHP 4.4 on an Apache 1.3 environment.

4.1 Data Set and Evaluation Metric

The experimental data comes from *MovieLens* which is a web-based research recommendation system (www.movielens.org). The data set contains 100,000 ratings of 1682 movies rated by 943 users (943 rows and 1682 columns of a *user-item matrix A*). These ratings were divided into two groups: 80% of the data (80,000 ratings) was used as a training set and 20% of the data (20,000 ratings) was used as a test set. Prior to evaluating the accuracy of the proposed method, a *user-item error matrix E* should first be constructed. Therefore, the training data set was further subdivided into training and testing portions, a matrix *E* was generated using a 5-fold cross validation scheme. After this process, a model (*an item-item trust matrix T*) for evaluating the method was created.

In order to measure the accuracy of the predictions, *mean absolute error* (MAE), which was widely used for the statistical accuracy measurements in the diverse algorithms [1, 2, 7] was adopted. The mean absolute error for user *u* is defined as:

$$MAUE(u) = \frac{\sum_{i\in I_u} | A_{u,i} - P_{u,i} |}{| I_u |}$$

where I_u is a item list of user u and $<A_{u,i}, P_{u,i}>$ is the actual/predicted rating pairs of user u in the test data. Finally, the MAE of all users in the test set is computed as:

$$MAE = \frac{\sum_{u=1}^{k} MAUE(u)}{K}$$

4.2 Parameter Tuning Experiments

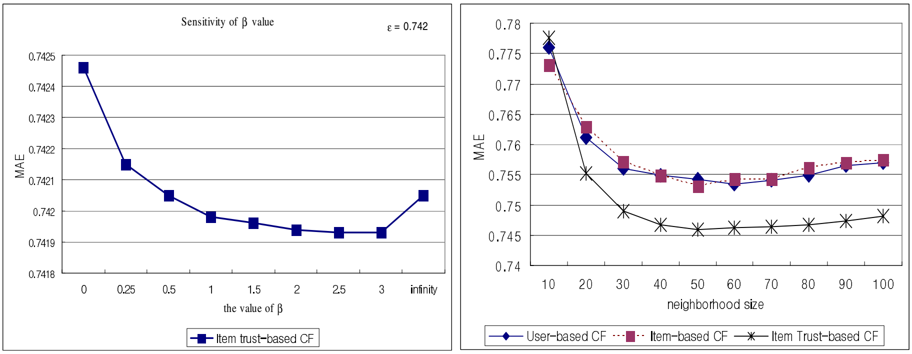
Prior to running the main experiment, the sensitivity of the two parameters: *item-item similarity* and β value, were first determined. In determining the sensitivity of these parameters, the training data set was focused on, which was further divided into two portions, 80% training and 20% testing. For parameter evaluation experiments, the full model size was used for model building, and k=30 was selected meaning the number of most similar items, for prediction generation.

Comparison of Similarity Algorithms. Prior to evaluating the item trust-based prediction method, a *user-item predicted matrix*, *P*, for calculating item confidence which is closely connected with the similarity algorithm, should first be built up. Thereby, we implemented diverse similarity algorithms such as correlation-based similarity, cosine-based similarity, adjusted cosine similarity as described in [2] and correlation-based similarity with inverse item frequency (*correlation+iif*) as described in [1]. And we compared them with cosine-based similarity with inverse item frequency (*cosine+iif*) as described in Section 3.1. For each similarity algorithms, the item-based Resnick measurement was used to generate the prediction. As seen from the results of Table 1, the prediction with the *cosine+iif* algorithm was generated, the prediction quality is improved, when compared to the other algorithms. Therefore, the cosine similarity with inverse item frequency is taken up in subsequent of experiments.

Table 1. Comparison of the prediction quality achieved by five different similarity measures

	cosine	cosine + iif	correlation	correlation + iif	adjusted cosine
MAE	0.74919	0.74248	0.75496	0.75242	0.76408

Sensitivity of β Value for Item Trust. As stated in Section 3.3, β is the parameter used for adjusting the relative weighting, where the similarity of items and the confidence of an item are important in the generation of an item trust. From the previous experiment, the error threshold ε for calculating the item confidence was set to be MAE of 0.742. Fig. 3(1) presents a variation in average MAE, by changing the β value. As a result, it can be observed that the quality of prediction improves as the β value is increased from 0 to 2, after 2, the curve tends to become flat. When β is set to infinity, the curve of the graph tends to rise. Hence, β=2.5 is selected as an optimal value for computing the item trust.



(a) (b)

Fig. 3. Sensitivity of parameter β for the item trust (a) and Comparison of prediction quality of user-based CF, item-based CF and Item trust-based CF (b)

4.3 Performance Evaluation

The performance evaluation is divided into two dimensions. The quality of the prediction based on item trust is first evaluated, and then the robustness of the prediction to the malicious ratings problem is evaluated. Once the optimal values of the parameters are obtained, the prediction quality of the proposed method is evaluated in comparison with the traditional user-based and item-based schemes.

Quality of the Prediction. The model size has significant impact on the prediction quality in a model-based approach [5]. However, the experimental result of the previous research in [2] demonstrates that a full model size obtains superior prediction quality than a small model size, although the time cost for building the model is greater. Therefore, in the prediction quality experiment, the full model size was used, and the number of item neighbors to be used for the online prediction generation was changed. The experimental results are depicted in Fig 3(b). It can be observed from the graph that the size of the neighborhood affects the prediction quality and the three methods demonstrate similar types of charts. The model-based approaches (item-based CF and item trust-based CF) elevate the prediction quality as the neighborhood size increases from 10 to 50, after this value, the quality decreased slightly. Likewise, a user-based CF improved until a neighborhood size of 60. The result demonstrates that, at all neighborhood size levels, except for a neighborhood size of 10, the proposed algorithm provides more accurate predictions than the traditional user-based and item-based algorithm. For example, when neighborhood size is 50, item trust-based CF obtains an MAE of 0.745, which is the best prediction quality, whereas item-based and user-based methods demonstrate an MAE of 0.753 and 0.754 respectively. However, the classic item-based scheme provides better quality in the event of a high sparsity level (neighborhood size =10).

Robustness on Malicious Ratings. For evaluating the robustness on fraud ratings, 10%, 20%, 30%, and 40% of malicious ratings were included in the training set, and the experiments were ran again using the full model size and a neighborhood size of 30. Table 2 summarizes the result of the experiment. In general, with the growth of malicious ratings, the prediction quality decreases, as can be seen from Table 1. However, the item trust-based CF shows the improved performance on all occasions, compared to traditional user-based and item-based CF. As the percentage of fraud ratings in a training data set increases, efficient improvement in performance can be obtained. Although the prediction quality is improved slightly in the case of 10% ratings being malicious, in the case of 40% ratings being malicious the proposed method achieves 5% improvement, compared to the other methods, respectively. As a result, the item trust-based CF brings 12% degradation in terms of the four cases in average, compared to an original rating set (0% malicious ratings set) whereas the average degradation of robustness is 15% for the user-based CF and 14% for the item-based CF.

Table 2. Robustness of user-based CF, item-based CF and item trust-based CF on fraud rating

malicious rations	User-based CF	Item-based CF	Item Trust-based CF
0 %	0.756	0.7572	0.7489
10 %	0.8042	0.8051	0.7954
20 %	0.8649	0.8601	0.8407
30 %	0.9542	0.9413	0.9184
40 %	1.0119	1.0059	0.9551

5 Conclusion and Future Work

Collaborative Filtering for Recommendations is a powerful technology for users to find information relevant to their needs. We have presented, in this paper, a novel approach to provide the enhanced prediction quality and to solve some of the limitation in traditional CF systems. And we propose a new method of building a model, namely *item-item trust matrix*, for CF-based recommender systems. The major advantage of the proposed approach is that it supports the protection against the influence of malicious ratings, or unreliable users. The experimental results demonstrate that the proposed method obtains significant advantages both in terms of improving the prediction quality and in dealing with malicious data sets as compared to traditional CF algorithms. However, there still remains a defect that the proposed method performs worse at a high sparsity level.

An ongoing area of current is a distributed recommender system [10, 12]. We are currently extending our algorithm to a personalized recommendation in a peer-to-peer environment or a social network. Therefore, we will further study the impact of using trust values, such as *web of trust* [6], and the technique of trust propagations.

References

1. Breese, J.S., Heckerman, D., Kadie, C.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In Proc. of the 14th Conf. on Uncertainty in Artificial Intelligence (1998) 43–52
2. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based Collaborative Filtering Recommendation Algorithms. In Proc. of the 10th Int. Conf. on World Wide Web (2001)

3. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. Riedl, J.: GroupLens: an open architecture for collaborative filtering of netnews. In Proc. of the ACM Conf. on Computer supported Cooperative Work (1994) 175–186
4. Lemire, D., Maclachlan, A.: Slope One Predictors for Online Rating-Based Collaborative Filtering. In Proc. of SIAM Data Mining (2005)
5. Deshpande, M., Karypis, G.: Item-based top-N recommendation algorithms. ACM Transactions on Information Systems, Vol. 22 (2004) 143–177
6. Massa, P., Avesani, P.: Trust-aware Collaborative Filtering for Recommender Systems. In Proc. of Int. Conf. on Cooperative Information Systems (2004)
7. Papagelis, M., Plexousakis, D., Kutsuras, T.: Alleviation the Sparsity Problem of Collaborative Filtering Using Trust Inferences. In Proc. of the 3rd Int. Conf. on Trust Management (2005) 224–239
8. O'Donovan, J., Smyth, B.: Trust in recommender systems. In Proc. of the 10th Int. Conf. on Intelligent user interfaces (2005) 167–174
9. Mobasher, B., Jin, X., Zhou, Y.: Semantically Enhanced Collaborative Filtering On the Web. Lecture Notes in Computer Science, Vol. 3209. Springer-Verlag, Berlin Heidelberg (2004) 57–76
10. Kim, H. J., Jung, J. J., Jo, G. S.: Conceptual Framework for Recommendation System based on Distributed User Ratings. In Proc. of the 2nd Int. Workshop on Grid and Cooperative Computing (2003)
11. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Analysis of recommendation algorithms for E-commerce. In Proc. of ACM'00 Conf. on Electronic Commerce (2000) 158–167
12. Miller, B. N., Konstan, J. A., Riedl, J.: PocketLens: Toward a personal recommender system. ACM Transactions on Information Systems, Vol. 22 (2004) 437–476
13. Schein, A. I., Popescul, A., Ungar, L. H.: Methods and Metrics for Cold-Start Recommendations. In Proc. of the 25th Int. ACM Conf. on Research and Development in Information Retrieval (2002)

Persuasive Online-Selling in Quality and Taste Domains

Markus Zanker[1,3], Marcel Bricman[2], Sergiu Gordea[1], Dietmar Jannach[1,3], and Markus Jessenitschnig[1]

[1] University Klagenfurt
9020 Klagenfurt, Austria
markus.zanker@uni-klu.ac.at
[2] KCI-group
9500 Villach, Austria
mbricman@kci-group.com
[3] ConfigWorks GmbH
9020 Klagenfurt, Austria
office@configworks.com

Abstract. 'Quality & taste' products like wine or fine cigars are one of the fastest growing product sectors in e-commerce. Online shops for these types of products require on the one side persuasive Web presentation and on the other side deep product knowledge. In that context recommender applications may help to create an enjoyable shopping experience for online users. The Advisor Suite framework is a knowledge-based conversational recommender system that aims at mediating between requirements and desires of online shoppers and technical characteristics of the product domain.

In this paper we present a conceptual scheme to classify the driving factors for creating a persuasive online shopping experience with recommender systems. We discuss these concepts on the basis of several fielded applications. Furthermore, we give qualitative results from a long-term evaluation in the domain of Cuban cigars.

1 Introduction

High complexity and search effort for the users are major obstacles for today's e-commerce sites that keep online conversion rates low and make potential customers restrain from buying. In traditional brick and mortar businesses sales persons fulfill an advising and complexity reducing function, but online customers many times still miss such a mediation between their personal requirements and the offered services and product portfolios.

Conversational recommender systems are successful applications of Artificial Intelligence that target this problem and ease customers in finding what they are looking for. However, recommender systems may not only be seen from a technical perspective in terms of their capabilities and functionality. Being the face towards the customer in an online-sales channel, there is also a social and

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 51–60, 2006.

emotional perspective that must be considered. In domains such as wine, cigars, or other luxury goods these aspects are of paramount importance. Comparable to their alive counterparts, virtual shop assistants may be intentionally designed to influence the online-customer's attitude or behavior in order to sell products and therefore they can be seen as a form of persuasive technology [1]. Dormann [2] discusses the emotional aspects of design and gives several examples of existing shop environments where customers might be actually more likely to buy. Guerini et al. [3] formalize persuasion models for intelligent interfaces that are based on beliefs, desires and intentions. In [4], a study has been conducted that investigated differences in people's perception of human and computer advice: Although results varied slightly depending on the culture and previous experiences of participants, the study showed that advice given by a computer was well accepted.

Sales advisory systems or virtual sales assistants are conversational recommender systems. They engage in an interaction with the customer and reach from simple value elicitation forms to more natural-language like sales dialogues. Depending on the underlying reasoning technology different variants of conversational recommender systems can be distinguished. Case-based recommender systems [5,6] employ enhanced similarity measures to retrieve those products that are most similar to the one specified by the user. Knowledge-based recommender systems [7,8] base their product proposals on deep domain knowledge and map the situational context of the customer onto technical product characteristics. Due to this explicit domain knowledge formulated as declarative business rules or constraints, recommendations can be accompanied with additional explanations for the user.

The third type of conversational recommender systems allows natural language discourses [9]. While the first two techniques allow only pre-structured interaction paths, the latter form can cope with natural language input of the customer. However, this additional flexibility is typically bought at the cost of limited product retrieval capabilities. For instance pre-defined querying interfaces allow more complex preference elicitation (e.g. selection from several choices and adding of restrictions) than what could be communicated to a natural language understanding system. The best known variants of recommender systems namely collaborative and content-based filtering systems [10,11] are in their pure form not conversational. Content-based recommender systems reason on preference models for users that are built from characteristics of products they liked in the past. Contrastingly, collaborative systems propose products to a user based on positive ratings from other users that are considered similar to him. Hybrid approaches [12,13] that combine several of these techniques and may also possess conversational capabilities have been developed recently.

In this paper, we discuss the persuasive power of conversational recommendation applications and present a conceptual scheme that emphasizes on the driving factors and their dependencies that influence user experience at online-shopping sites (Section 2). In Section 3 we give qualitative results from a long running evaluation that indicates commercial benefits from the deployment of recommender applications. Finally, we summarize the contribution and make an outlook on future work.

2 Persuasive Recommendation Scheme

The field of persuasive computing technologies [1] is still in its infancy. It lies at the intersection of technology and tools on the one side and social acting and social dynamics on the other side. Computing applications that influence people's attitudes or behaviors are also in the center of ethical discussions [14]. In our work we set the focus on effective sales applications for the Web. By understanding the social and personality traits of successful salespersons and their interaction and selling techniques, valuable insights for the functioning of sales agents in the Web can be gained. Selling has always been closely related to social behavior and persuasion in a wider sense. Building effective virtual salespersons for an e-Commerce environment that provides advisory service to their users and helps shop owners to be successful in monetary terms can therefore benefit from following the principles of persuasive technologies. Up to now only few work has been elaborated on these issues. In [15] a stepwise process that characterizes a seductive experience with a software product is described: They differentiate phases such as diverting the attention, making a promise or fulfillment beyond expectations. Komiak et al. [16] report on a study where they compared trust in virtual salespersons with trust in their human counterparts. Some of the interesting findings were that virtual salespersons were attributed to be slightly more trustworthy than human ones. Control over the process and benevolence (due to the fact that they do not earn a commission) were among the strongest perceived trust building components for virtual salespersons.

As each of the contributions in this area focuses only on isolated aspects of recommendation in the context of persuasiveness, we propose a conceptual framework that structures the influential factors and their dependencies. As sketched in Figure 1 online selling centers around customer, product and process. The basic abstract features with respect to recommender systems comprise: Personalization, personification, community, objectivity and complexity reduction. They positively influence higher level concepts such as user involvement and credibility and thus believability of the system itself. On the third layer reciprocity and conditioning are situated. They signify the actual ability of a recommender system to influence users behavior and affect their attitudes. The final stage leads to adopted suggestions and a committed sale. In the following we will shortly discuss each of the involved concepts and illustrate them with examples from the domains of wine and cigars. **Personalization** denotes the adaptation of the system behavior and its outcome towards the situational as well as the preferential and requirements context of the user [17]. In [18] different personalization strategies supported by the Advisor Suite on both the level of presented content and on the interaction process itself are elaborated. When buying wine online, users like to have the choice between different ways of stating their preferences. For instance selecting product parameters like region, year or grape variety on the one

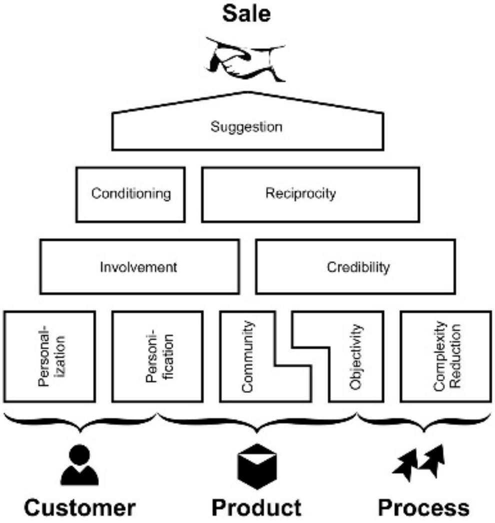


Fig. 1. Building blocks for persuasive sales recommenders

hand and describing taste characteristics like sweetness or fruitiness on the other hand. When proposing products to customers, providing them also additional product details and some form of explanation proves extremely helpful [16].

Personification is about giving a life-like identity to virtual sales agents. Full resemblance to humans is however not necessary. Virtual personalities can be created by using a relatively small inventory of traits [19]. Nevertheless, making the character look attractive is not an easy task. Rules of thumb like symmetry, broken symmetry, golden cut or color psychology might still produce an unfavorable outcome. Therefore, extensive user tests on the characters are necessary. [20] showed that attractiveness of virtual characters correlates positively with purchase likelihood [20]. In Figure 2 the cigar advisor *Mortimer* is depicted.

Community features of recommender systems are predicted to have a big potential in the future [21]. Computer games which hold the forefront of technology development are rapidly moving towards building online communities. Ratings and comments to products that can be seen by all users or topic centric discussion fora have only been the beginning. In the future we might see social networking systems or virtual sales rooms where basic social behavior between customers becomes possible.

Objectivity is mutually affected by community aspects. Opinions from other customers are generally appreciated as they are assumed to be independent from the shop owner. However, recommender systems can foster their perceived objectivity by offering a high degree of product information, explaining pros and cons for proposals and allowing testimonials and comments from other users. For instance the virtual sommelier *sem* reasons on product specific explanations that point the customer explicitly on the dryness of a wine if its acidity level is high and its sugar level low.

Conversational recommender systems achieve **Complexity reduction** for customers by mediating between the situational and preferential context of the client and the technical details of the product domain. Another crucial factor is the tool complexity of the system itself and its perceived usability.

Involvement and **credibility** may both be induced by the underlying building blocks. Involving users signifies reducing the distance between them and the system itself [22]. In our case community building as well as personalized customer treatment are prerequisites for user involvement. Personification through virtual characters stimulates users involvement by enabling them to identify themselves with the system. Tseng and Fogg [23] circumscribe the credibility of a system with the synonym *believability*. It addresses the basic question: May users believe the information given? Swana et al. [24] provide a comprehensive review on the issue of customer trust in salespersons. In virtual environments the factors objectivity and community building may positively affect this issue. Furthermore, knowledge-based recommender systems have the advantage of incorporating the knowledge of the human expert, that guides and advises (complexity reduction). Communicating explicitly the competency of a real-world sommelier behind the virtual character *sem* is a strategy taken by the online wine platform.

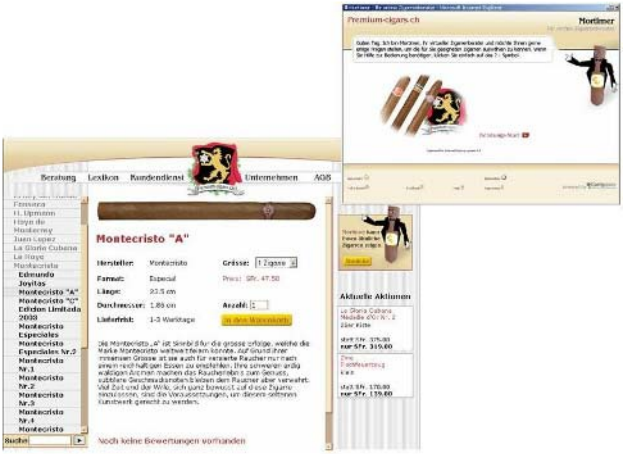
Customers that feel involved relate to computers and applications like they relate to other humans. Therefore, humans would accept praise by a computer, e.g. when the virtual shop assistant compliments the user on its excellent choice. Thus introducing **conditioning** into conversational dialogues is one of the final steps towards persuasiveness of recommendations. Closely related to the issue of conditioning is **reciprocity**. Fogg and Nass [25] conducted experiments to investigate how users can be motivated to change their behavior by a computer. In online sales situations keeping the user at the Web site and in the process is of paramount importance. Thus, credibility of the system and user involvement are the underlying factors that foster loyalty of users.

Suggestion is on top of our conceptual scheme and stands for the ability of virtual sales agents to successfully close the deal when needed. Patience is not widespread among online users, therefore a balance between interaction length and suggestive propositions is crucial for conversational recommendation agents to influence people's minds.

In the following we give practical evidence from the implementation of some of these abstract concepts in an e-Commerce platform for Cuban cigars and conduct an analysis of actual sales figures.

3 Example

Most reported experiments in the domain of recommender systems perform an off-line analysis on an historical data-set [26]. So, for instance the predictive accuracy of collaborative filtering algorithms [26] or the session length for different retrieval strategies are measured. The evaluation effort is considerably higher when conducting live-user trials with an application prototype like [27] or



Guten Tag. Ich bin Mortimer, Ihr virtueller Zigarrenberater und möchte Ihnen gerne einige Fragen stellen, um die für Sie geeigneten Zigarren auswählen zu können. Wenn Sie Hilfe zur Bedienung benötigen, klicken Sie einfach auf das ? - Symbol.

Montecristo "A"

Hersteller: Montecristo
Format: Especial
Länge: 23.5 cm
Durchmesser: 1.88 cm
Lieferfrist: 1-3 Werktage
Grösse: 1 Zigarre
Preis: SFr. 47.50
Anzahl: 1

Die Montecristo „A" ist Sinnbild für die grosse Erfolge, welche die Marke Montecristo weltweit feiern konnte. Auf Grund ihrer immensen Grösse ist sie auch für versierte Raucher nur nach einem reichhaltigen Essen zu empfehlen. Ihre schweren erdig waldigen Aromen machen das Raucherlebnis zum Genuss, subtilere Geschmacksnoten bleiben dem Raucher aber verwehrt. Viel Zeit und der Wille, sich ganz bewusst auf diese Zigarre einzulassen, sind die Voraussetzungen, um diesem seltenen Kunstwerk gerecht zu werden.

Noch keine Bewertungen vorhanden

Fig. 2. Screenshot of cigar shop

operating a real system like Entree that served as a restaurant guide for attendees of a convention in Chicago [12].

Moreover, our research question is about the capability of recommender systems to influence peoples mind when interacting with recommender systems. Therefore we even chose a real-world setting to analyze the effect of a conversational recommender on buying behavior and sales figures. In the following we report about a Web shop for fine Cuban cigars that introduced the virtual shop assistant *Mortimer* on the World non smokers day (May, 31^{st}) in 2003. Figure 2 gives a screenshot of the Web shop environment and the virtual sales assistant. The conversational recommender system is based on the Advisor Suite framework [8] that was extended to include additional support for the integration of virtual characters. Next to personification, also the personalization and complexity reduction function was targeted by the sales recommender system. The shop owner addressed community building by introducing testimonials of registered users. To ensure objectivity the advisor system generates specific explanations on products upon request that provide pros as well as cons with respect to the user preferences. While being strong on basic functionalities, implementation of features that support the higher level concepts like conditioning and involvement is left for future work.

For evaluation, we were given the opportunity to analyze the sales figures for a period starting in January 2002 until November 2004. The product range comprises approximately 115 different models of Cuban cigars from 18 different manufacturers with *Cohiba*, *Montecristo* or *Romeo y Julieta* being the most prominent ones. The assortment of products and their prices remained basically stable over the whole evaluation period.

The research hypothesis was as follows: **Does advice given by a virtual shop assistant affect the buying behavior of online shoppers?**

Therefore, we separated sales records into two periods, one before Mortimer went live (01/2002 - 05/2003) and one afterwards (06/2003 - 11/2004). For each

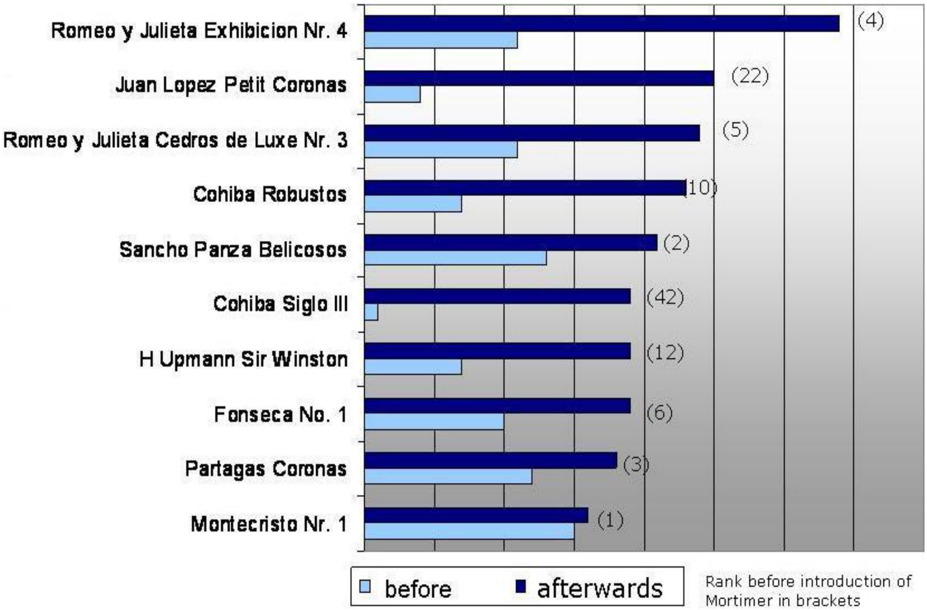


Fig. 3. List of top ranked products

period we computed a separate top ten product list. The result confirmed the initial guess of the shop owner, namely that customers are more likely to order cigars produced by less prominent manufacturers then before. In the period before Mortimer, mostly the prominent makes like *Cohiba* or *Montecristo* have been ordered, while in the period afterwards 'no-names' like *Juan Lopez Petit Coronas* entered the top ten list (e.g. rank 2 compared with rank 22 in the period before). Figure 3 gives the top ten ranks for the period afterwards from top to bottom. The bars signify actual sales in pieces for the period before and afterwards. No numbers are given for reasons of confidentiality, but the ratios between bars are in accordance with the absolute numbers. The figures in brackets next to the bars of each cigar model correspond to the rank in the period before, e.g. the top ranked model in the period afterwards *Exhibicion Nr. 4* by *Romeo y Julieta* was previously ranked on 4^{th} position and the top ranked model in the period before *Montecristo No. 1* fell back to rank number 10. Evaluations of shorter subperiods did not result in significant changes. In a next step we wanted to drill down on single interactions of users and relate the interaction data collected by the sales advisor with the ordering records of the web shop. However, online customers were anonymously browsing the shop until identifying themselves for ordering. As many users receive a dynamic IP-address from their internet provider and sales acts typically last over a longer time span (e.g. customers browse and collect some information and come back later to finish an order) we were not successful in joining tuples from different tables and log files. However, a single question was already part of the ordering

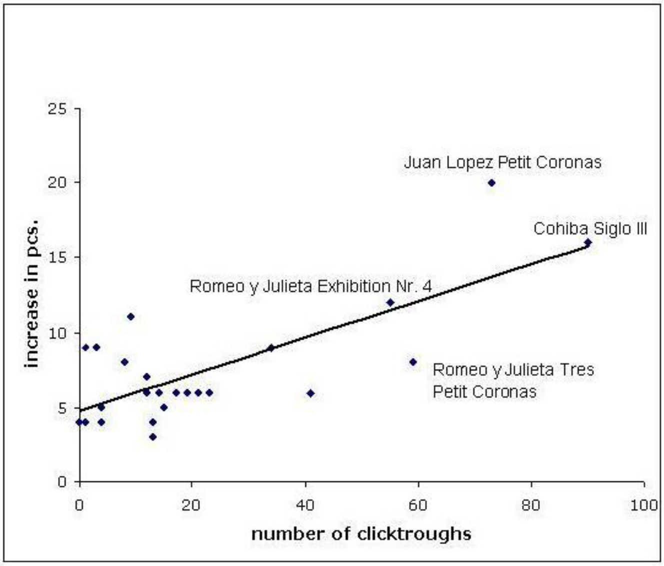


Fig. 4. Correlation between clickthroughs and additional sold items

form where users could explicitly place a tick if Mortimer helped them placing this order. In more than 26% of all sales customers agreed that Mortimer did help them to make a decision. This does not seem a lot at first sight, but given the fact that many cigar smokers will just order their favorite model without conversing with a recommender, the figure is quite promising.

In addition, we compared how Mortimer's recommendations correlate with the increase in sold items of different cigar models. On the abscissa we measure the number of clickthroughs, which denotes how often an item has been recommended and was then examined in detail by the user. As sketched in Figure 4 cigar models that greatly improved their rank are also among the items that have been recommended most often. We also investigated why cigars like *Juan Lopez Petit Coronas* or *Cohiba Siglo III* were recommended so often and thus became so popular among customers. It turned out that in specific situations for instance when users identified themselves as novices without any smoking experiences these models have been proposed due to their taste and smoking duration. Recommendations are always accompanied by explanations and further product details as already noted in the previous section. Although overall correlation between recommendations and sales for this example is below 0.4, it still qualitatively supports the following conclusions: Recommender systems do affect the shopping behavior of users and their advice is accepted by them.

4 Conclusions

In this paper we discussed concepts for persuasive technologies and related them to the development of conversational recommender systems. We sketched a conceptual scheme that builds on basic functionalities of recommender agents like personalization or objectivity and finally leads to suggestions that are persuasive in nature, i.e. affect peoples actual buying decisions. We illustrate the concepts with examples from systems deployed in quality & taste domains and provide results from an evaluation conducted with data from a Web shop for Cuban premium cigars.

Future work will lead us to further evaluation scenarios where key ingredients for persuasive sales advisors will be analyzed in greater detail and their effects and dependencies will be evaluated.

References

1. Fogg, B.J.: Persuasive Technologies. Communications of the ACM **42(5)** (1999) 27–29
2. Dormann, C.: Designing electronic shops, persuading customers to buy. In: Proceedings of the 26^{th} Euromicro Conference. (2000) 140–147
3. Guerini, M., Stock, O., Zancanaro, M.: Persuasion Models for Intelligent Interfaces. In: Proceedings of the IJCAI Workshop on Computational Models of Natural Argument. (2003)
4. Yvonne Waern and Robert Ramberg: People's perception of human and computer advice. Computers in Human Behavior **12(1)** (1996) 17–27
5. Ricci, F., Werthner, H.: Case base querying for travel planning recommendation. Information Technology and Tourism **3** (2002) 215–266
6. O'Sullivan, D., Smyth, B., Wilson, D.: Understanding case-based recommendation: A similarity knowledge perspective. International Journal of Artificial Intelligence Tools (2005)
7. Burke, R.: Knowledge-based recommender systems. Encyclopedia of Library and Information Systems **69** (2000)
8. Jannach, D.: Advisor suite - a knowledge-based sales advisory system. In: European Conference on Artificial Intelligence - ECAI 2004. (2004)
9. Lucente, M.: Conversational interfaces for e-commerce applications. Communications of the ACM **43(9)** (2000) 59–61
10. Balabanovic, M., Shoham, Y.: Fab: Content-based, collaborative recommendation. Communications of the ACM **40(3)** (1997) 66–72
11. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Analysis of recommendation algorithms for e-commerce. In: ACM Conference on e-Commerce (EC). (2000) 158–167
12. Burke, R.: Hybrid recommender systems: Survey and experiments. User Modeling and User-Adapted Interaction **12(4)** (2002) 331–370
13. Adamavicius, G., Tuzhilin, A.: Towards the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering **17(6)** (2005)
14. Berdichevsky, D., Neuenschwander, E.: Toward an Ethics of Persuasive Technologies. Communications of the ACM **42(5)** (1999) 51–58
15. Khaslavsky, J., Shedroff, N.: Understanding the Seductive Experience. Communications of the ACM **42(5)** (1999) 45–49
16. Komiak, S., Wang, W., Benbasat, I.: Comparing Customer Trust in Virtual Salespersons With Customer Trust in Human Salespersons. In: Proceedings of the 38^{th} Hawaii International Conference on System Sciences (HICSS). (2005)

17. H. H. Sung: Helping Customers Decide through Web Personalization. IEEE Intelligent Systems **17(6)** (2002) 34–43
18. Jannach, D., Kreutler, G.: Personalized User Preference Elicitation for e-Services. In: Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE). (2005)
19. Nass, C.I., Moon, Y., Fogg, B.J., Reeves, B., Dryer, D.C.: Can computer personalities be human personalities? International Journal of Human-Computer Studies **43** (1995) 223–239
20. Suzuki, S.V., Yamada, S.: Persuasion through overheard communication by life-like agents. In: Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT). (2004) 225–231
21. Tedjamulia, S.J.J., Olsen, D.R., Dean, D.L., Albrecht, C.C.: Motivating Content Contributions to Online Communities: Toward a More Comprehensive Theory. In: Proceedings of the 38^{th} Hawaii International Conference on System Sciences (HICSS). (2005)
22. Wilson, S., Bekker, M., Johnson, P., Johnson, H.: Helping and Hindering User Involvement - A Tale of Everyday Design. In: Proceedings of Conference on Human Factors in Computing Systems (CHI). (1997)
23. Tseng, S., Fogg, B.J.: Credibility and Computing Technology. Communications of the ACM **42(5)** (1999) 39–44
24. John E. Swana and Michael R. Bowersa and Lynne D. Richardson: Customer Trust in the Salesperson: An Integrative Review and Meta-Analysis of the Empirical Literature. Journal of Business Research **44(2)** (1999) 93–107
25. Fogg, B.J., Nass, C.: How Users Reciprocate to Computers: An experiment that demonstrates behavior change. In: Proceedings of Conference on Human Factors in Computing Systems (CHI). (1997)
26. Jung, S., Harris, K., Webster, J., Herlocker, J.: Serf:integrating human recommendations with search. In: Thirteenth Conference on Information and Knowledge Management (CIKM 2004). (2004)
27. McCarthy, K., Reilly, J., McGinty, L., Smyth, B.: Experiments in Dynamic Critiquing. In: Proceedings of the Intelligent User Interfaces Conference (IUI). (2005) 175–182

Proviado – Personalized and Configurable Visualizations of Business Processes*

Ralph Bobrik[1], Thomas Bauer[2], and Manfred Reichert[3]

[1] Dept. Databases and Information Systems, University of Ulm, Germany
ralph.bobrik@uni-ulm.de
[2] DaimlerChrysler Research & Technology, REI/ID, Ulm, Germany
thomas.tb.bauer@daimlerchrysler.com
[3] Information Systems Group, University of Twente, The Netherlands
m.u.reichert@cs.utwente.nl

Abstract. A monitoring component is a much-needed module in order to provide an integrated view on system-spanning and cross-organizational business processes. Current monitoring tools, however, do not offer adequate process visualization support. In particular, processes are always visualized in the way they were drawn by the process designer. This static approach is by far not sufficient when dealing with more complex scenarios where different user groups usually have different perspectives on processes and related data. In such an environment different views and personalized visualizations have to be provided. In the Proviado project we are developing a framework for realizing flexible and adaptable visualizations of business processes whose data may be scattered over multiple information systems. In this paper we focus on personalization and configuration issues, and we show how process visualizations can be adapted automatically, e.g., by applying different notations for different user groups or by altering the appearance of visualized elements depending on their execution state. For this purpose we define a visualization model which maintains all required visualization parameters.

1 Introduction

In order to streamline their way of doing business, today's companies have to support a number of business processes (BP) involving different partners, departments, and actors. In this context we have seen an increasing adoption of BP management technologies as well as emerging standards for BP orchestration (e.g., BPEL4WS) and BP choreography (e.g., WS-CDL) [1]. These technologies and standards enable the definition and execution of the operational processes of an enterprise. In connection with Web Service technology, in addition, the benefits of BP automation and BP optimization from within a single enterprise can be transferred to cross-organizational business processes as well.

A BP monitoring component is a much-needed module, particularly if process data are scattered over distributed, heterogeneous information systems. It has

* This work has been funded by *DaimlerChrysler Research & Technology.*

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 61–71, 2006.

to provide comprehensive visualization support for both model and instance data. A *process model* defines the BP activities, the control and data flow between them (e.g., represented by control/data edges connecting activities among each other or linking activities with data elements), and other process aspects (e.g., resources). A *process instance*, in turn, is executed on basis of a particular process model, but comprises additional run-time information to be displayed (e.g., activity states or application data). An example is depicted in Fig. 1.

One major shortcoming of current BP monitoring software is the static way in which business processes are visualized. Usually, a process model is displayed to users in exactly the same way as designed (or painted) by the process engineer at build time. Any adaptation of model contents (e.g., hiding automated activities) may cause major efforts for re-drawing the process model, particularly if it comprises dozens or hundreds of activities, data objects, and other graphical elements. In most cases, it is even not possible to adapt the graphical appearance of a process model to user preferences at runtime. What is needed is a runtime-adaptable BP visualization, which allows to suppress skipped execution branches, to hide certain process aspects (e.g., business documents, IT systems), or to only display activities belonging to a certain role. Further, in multi-user environments, BP visualizations should be adaptable to the preferences of process participants, e.g., allowing them to apply different notations for the same process or to vary colors and symbols for the different process elements.

In the Proviado project [2] we are developing a sophisticated framework for such adaptive and configurable BP visualizations. Areas of interest include the (semantic) integration of distributed, heterogeneous process data and their defragmentation, the provision of mechanisms for creating (dynamic) process views (e.g., by applying graph aggregation/graph reduction techniques), the automated layouting of process graphs (e.g., after changes), the use of different notations for a business process, and the provision of personalized process visualizations. In this paper we focus on configuration issues, i.e., we deal with the question how to (dynamically) configure, determine, and adapt the graphical appearance of process elements. We introduce a *visualization model* (VisModel) for this purpose, which allows for the high-level and user-friendly configuration of visualization parameters for a variety of process elements. The design of this model has been non-trivial, since it must capture different kinds of visualization parameters in a user-comprehensible and maintainable way. Examples of needed parameterizations include view definitions, links to real-time process data, layout strategies,

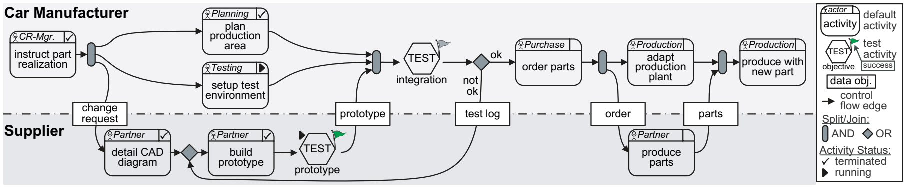


Fig. 1. Interorganizational Change Management Process (Realization phase)

preferred process notations, etc. We have elaborated the developed visualization model in several case studies in the automotive domain and implemented a powerful proof-of-concept prototype. Together with the above mentioned features the visualization model forms the key for the automated and dynamic generation of personalized process visualizations. By detaching process information from its visualization we achieve a strict separation of content and presentation, a well known approach from other areas like Web design, content management, or software development.

Section 2 starts with an example followed by the description of the requirements for realizing personalized process visualizations. Our solution approach is sketched in Section 3, and Section 4 describes our proof-of-concept prototype. In Section 5 we discuss related work. Section 6 concludes with a summary and an outlook on future research.

2 Configuration of Process Visualizations

2.1 Example

A simplified example of a cross-organizational process is depicted in Fig. 1. It shows an extract of a change management process from the automotive domain: After having decided that a certain part of a car (e.g., an electrical control unit) has to be modified, the implementation of the change is started. Related duties are then split among the car manufacturer and the part supplier. While the construction and production of the new part is in charge of the supplier, the car manufacturer must ensure that the car production facilities are adapted accordingly. For this purpose the supplier delivers a first prototype of the new part. If this prototype complies with the specifications and passes all tests ("integration tests"), it will be integrated in the production process. In order to keep track of this process a monitoring component providing site-specific visualizations of processes and related data is needed. The vision is to be able to generate adapted and personalized visualizations from given process data (see Fig. 1 and Fig. 2).

2.2 Requirements

In order to better understand the requirements for the design of our VisModel we elaborated real cases from the automotive sector. Requirements for the configuration of process visualizations are depicted in Table 1 (for details see [2]). Other requirements related to the overall visualization component (e.g., integration of process models and related runtime data, visualization of processes in different forms) can be found in [2].

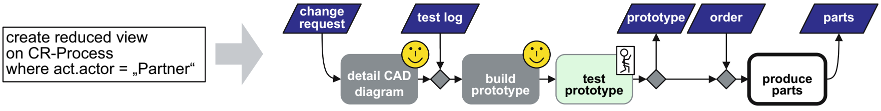


Fig. 2. Supplier's view on the process (from Fig. 1)

Table 1. Requirements for the configuration of process visualizations

Req. 1	Ability to use arbitrary symbols for visualizing a process element
Req. 2	Selection of visualization symbols based on the attribute values of the process elements
Req. 3	Precise rules for dealing with conflicting instructions regarding the visualization of process elements
Req. 4	Personalization of visualizations by adapting colors, fonts, symbols, etc.
Req. 5	Accessibility of process visualization via easy to maintain (Web) clients
Req. 6	Easy creation of new process notations and symbols
Req. 7	Easy implementation and integration with our overall architecture for BP visualization support

Req. 1 reflects the mentioned concept of separating content and presentation. In order to personalize process visualizations we must be able to easily adapt the symbol used for a certain process element. Among other things we have to specify at a high level which symbols shall be applied under which conditions. Generally, the graphical appearance of a process element depends on its properties (cf. Fig. 3). As an example consider process activities. By default we might want to represent activity nodes by a square with rounded edges. However, for activities of type *"testing"* this default representation should be replaced by a special symbol reflecting the test result with a colored flag (cf. Fig. 1). Or at the process instance level we might want to color activities depending on their state. Generally, arbitrary process data (i.e. process element attributes, instance and application data) may be used to determine the appearance of process elements. Additional complexity arises from the necessity to specify the format of a symbol dependent on process attributes (Req. 2), which are not associated with the current process element, but connected with another element via edges. Depending on the name of the actor working on an activity, for example, the color of the activity node may have to be altered, facilitating recognition of tasks belonging to the same actor. Altogether, for a complex process and its visualization this leads to a large number of dependencies or rules, describing the applicability of symbols and formats. In this scenario it is not far-fetched that two rules contradict each other. For example, let Rule 1 specify a red color for activities performed by a particular user. In contrast Rule 2 may state that running activities shall be emphasized with green color. Then it may occur that an activity fulfills both rules and it is not clear whether the activity shall be drawn red or green. This kind of conflicts between formatting instructions should be resolved in order to achieve consistent process drawings (Req. 3).

Adapted process visualizations must be made available to users as easy as possible and deployment efforts should be minimized (Req. 5). In particular, access to process information via Web browsers should be supported. In order to achieve this we use SVG (Scalable Vector Graphics) as format for displaying processes. Usually respective Web browser plug-ins are available and installed at the client side. Further, frameworks for supporting SVG on the server-side exist as well (Req. 7). For the task of generating process visualizations standard

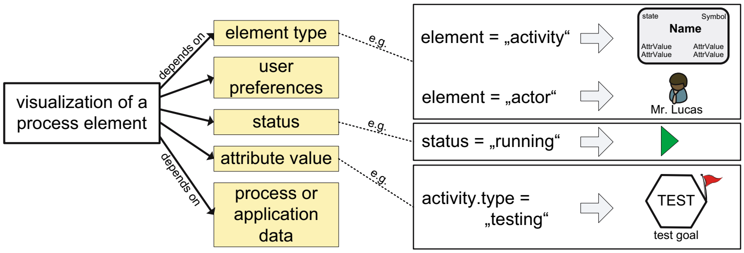


Fig. 3. Dependencies of data determining presentation

XML-based technologies (like XSLT or sXBL) allow to convert arbitrary XML data structures into SVG [3]. However, the direct application of these technologies would contradict our goal to separate content and presentation because of their complex syntax. In particular, graphical aspects are mixed with the logic for combining different templates, which results in high maintenance costs.

3 Visualization Model

Key component for specifying and maintaining visualization parameters is the Visualization Model (VisModel), whose entries are organized in an XML-based tree-structure. Fig. 4 depicts the role of this model in our overall approach for generating personalized process visualizations. The VisModel represents a logical view on the parameters for this visualization procedure. This includes, for instance, a representation of the process model to be displayed, an optional view definition reassembling the process model, a definition of the notation to be used, graphical settings regarding the appearance of process elements (e.g., colors, fonts, etc.), and information needed to access workflow or application data at runtime. In order to realize a particular process visualization we use one VisModel. Consequently, if different visualizations of a process are desired, logically, multiple VisModels have to be created. Note that the information needed for this can be gathered partially from existing information (e.g., reusing models capturing visualization profiles of a particular user group).

Fig. 4 shows the steps (S0–S3) necessary to automatically generate a process visualization. Starting point is an "integrated" process model, which correlates (fragmented) process data from different source systems in a harmonized way. First, we restrict this visualization content to that information needed by the user (S0). This is realized by a view component which applies aggregation and reduction techniques to process models (cf. Fig. 2). Step S0 is followed by formatting steps S1–S3: S1 fixes the graphical symbols designed for the different process elements. Doing so we consider information from the visualization model; S2 fills graphical symbols with real attribute values related to the process model or process instance to be displayed; within S3 formatting parameters are customized to user preferences, e.g., by coloring the process visualization in accordance to cooperate identity guidelines.

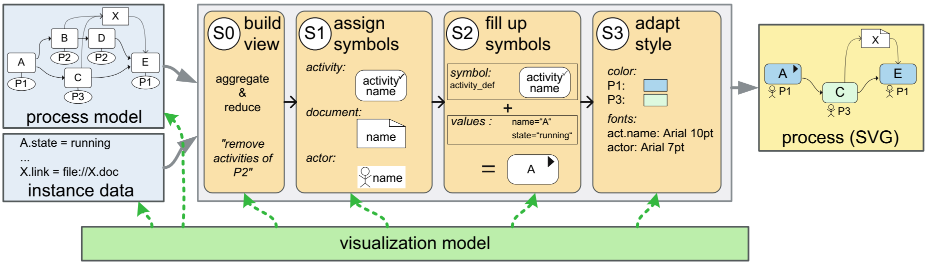


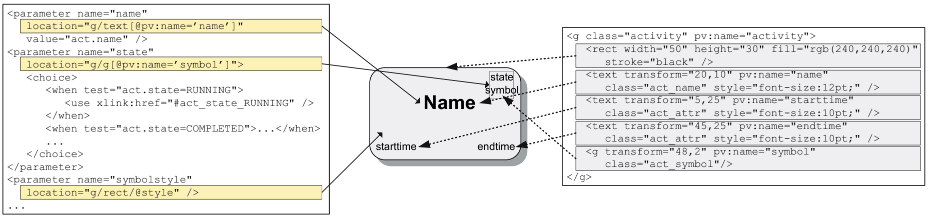
Fig. 4. Role of the Visualization Model in generating a process visualization

3.1 Template Mechanism

To enable the flexible configuration of the used process notation, we introduce a sophisticated template mechanism. Key design criteria have been Req. 1–3, with the reuse of existing templates in mind. The mechanism is subdivided in two parts: the description of the symbols and definition of their usage.

Describing Symbols. A template definition consists of three specification parts (cf. Fig. 6a): 1) input parameters of the template, where references to process elements are handed over; 2) representation of the symbol in SVG; 3) parameters (e.g. name of activity, activity state, starting time, etc.) to be filled with process data values. As mentioned, we adopted SVG as format for defining graphical symbols because of its XML-based syntax and the general advantages of vector graphics over raster graphics. This also allows for the easy definition of process symbols by using off-the-shelf SVG editors. In our approach each template defines exactly one symbol with its graphical characteristics (e.g. shapes and text areas). The text areas (i.e., parameters of the template) are filled by concrete values from the process model/instance to be displayed. Within the parameter sections, XPath expressions (relative to the SVG-symbol root) are used to describe the location of the corresponding text area (`location` attribute). As process data values may need to be transformed before presenting them to the user (e.g. converting an internal date format into a standard format; cf. Fig. 6b) the `value` attribute may comprise of code in a scripting language (e.g. JavaScript). Via these scriptlets it is further possible to access all kind of process data and to arrange it using arbitrary scripting functions (cf. parameter `endtime` in Fig. 6 a). For expressing special formatting options dependent on arbitrary process attributes, if-then-else or choice structures may be used. Hereby the evaluation conditions are expressed by also using JavaScript.

Fig. 5 shows an example for an activity template. The right side depicts the symbol definition based on SVG. On the left, corresponding parameter definitions are shown. Among other things they illustrate the mechanisms used to reference the locations of the data values inside the symbol. A choice construct is used to determine the correct process state symbol for activity nodes.



```
<parameter name="name"
   location="g/text[@pv:name='name']"
   value="act.name" />
<parameter name="state"
   location="g/g[@pv:name='symbol']">
   <choice>
      <when test="act.state=RUNNING">
         <use xlink:href="#act_state_RUNNING" />
      </when>
      <when test="act.state=COMPLETED">...</when>
      ...
   </choice>
</parameter>
<parameter name="symbolstyle"
   location="g/rect/@style" />
...
```

```
<g class="activity" pv:name="activity">
   <rect width="50" height="30" fill="rgb(240,240,240)"
      stroke="black" />
   <text transform="20,10" pv:name="name"
      class="act_name" style="font-size:12pt;" />
   <text transform="5,25" pv:name="starttime"
      class="act_attr" style="font-size:10pt;" />
   <text transform="45,25" pv:name="endtime"
      class="act_attr" style="font-size:10pt;" />
   <g transform="48,2" pv:name="symbol"
      class="act_symbol"/>
</g>
```

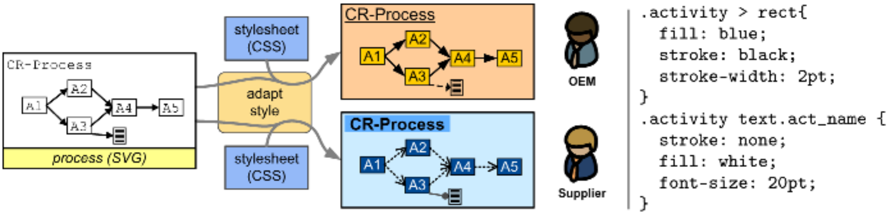
Fig. 5. Template mechanism: Definiton of a symbol

```
<template id="default_act">
  <!- input section ->
  <inputs>
    <input variable="act" type="activity">
      <descr>activity node</descr>
    </input>
  </inputs>
  <graphic>
    <!- symbol section (SVG) ->
    <symbol>
      <g class="activity" pv:name="activity">
        ...
      </g>
    </symbol>
    <!- parameter section ->
    <parameter name="name"
      location="g/text[@pv:name='name']"
      value="act.name" />
    ...
    <parameter name="endtime"
      location="g/text[@pv:name='endtime']"
      value="formatDate(act.end,'dd/mm/yyyy')"/>
  </graphic>
</template>
```

```
<if test="self.type=ACTOR">
  <template id="actor">
    <inputs>
      <input name="actor" value="self"/>
    </inputs>
  </template>
</if> <if test="self.type=ACTIVITY">
  <choose>
    <when test="self.type='testing'">
      <template ref="testing_act">
        <inputs>
          <input name="act" value="self"/>
        </inputs>
      </template>
    </when>
    <otherwise>
      <template ref="default_act">
        <inputs>
          <input name="act" value="self"/>
        </inputs>
      </template>
    </otherwise>
  </choose>
</if>
```

Fig. 6. (a) Definition of templates (b) Usage of templates

Defining Usage. Having defined a set of templates the next challenge is to specify under which conditions these templates shall be applied. Main complexity in this context is to define the correlations between process elements and available templates (cf. Fig 3). As an example consider the process from Fig. 1 where we want to use a rectangle with rounded edges for displaying activities and a special symbol for representing "test activities". We first considered using logic rules for this task, but withdrew this idea. First, we would have obtained a large number of rules to be maintained. Second, specification of such rules would have been a complex task (e.g., precise conditions for firing rules would have had to be specified). Third, rules are not guaranteed to be conflict-free; i.e., there might be two rules, one defining a red background color and the other specifying a green color. A general conflict resolution strategy for this case would be very complex to realize. The solution we have chosen instead is depicted in Fig. 6b. Using "if-then-else"-like statements together with a first-occurrence-wins policy, it is ensured that the template to be applied can be determined unambiguously at runtime. The algorithm we developed in this context traverses all elements of the process model and assigns the corresponding symbols to them.



```
.activity > rect{
  fill: blue;
  stroke: black;
  stroke-width: 2pt;
}
.activity text.act_name {
  stroke: none;
  fill: white;
  font-size: 20pt;
}
```

Fig. 7. Use of stylesheets for adapting format parameters

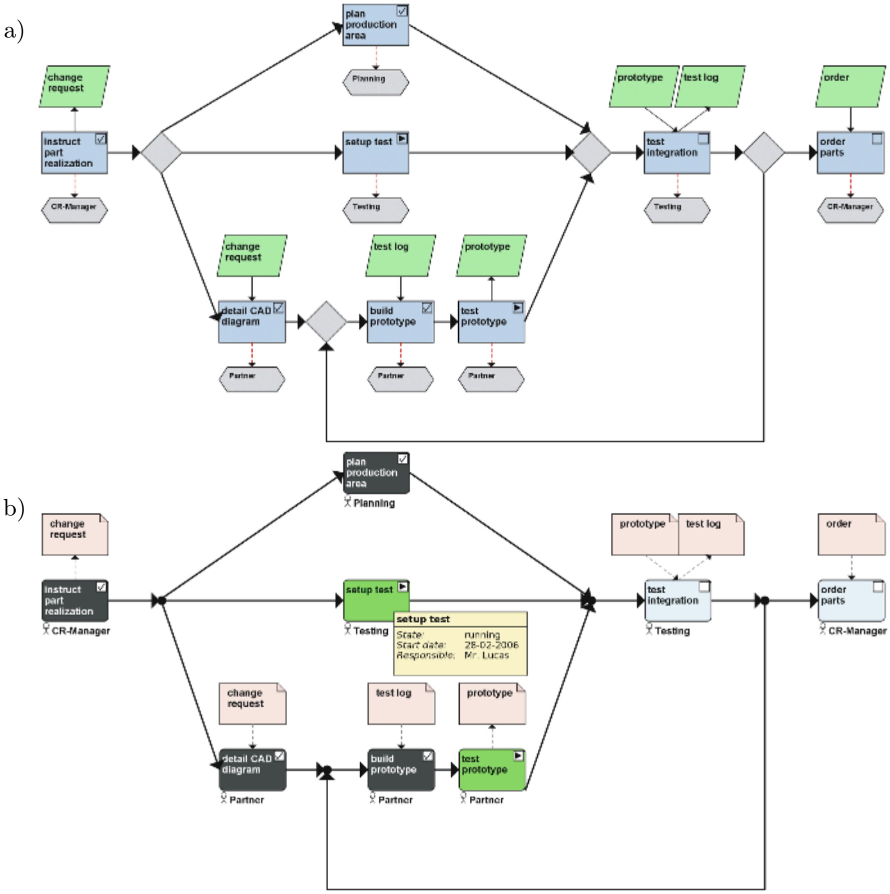
3.2 Formatting a Process Visualization

The task of formatting a process visualization is subdivided into 3 steps (cf. Fig. 4). In Section 3.1 we already explained how to describe unambiguously which symbol has to be used for visualizing a certain process element. This definition is interpreted in Step S1 where symbol templates are assigned to the process elements. In Step S2 the parameter values of the templates are calculated according to the scriptlets contained in the templates. Consequently, the placeholders are substituted by concrete values from the process model (instance). During Step S3 graphical attributes of the process elements are adapted according to users' preferences (cf. Fig 7). In this step, colors, fonts, line styles, etc. can be modified using Cascading Style Sheets (CSS). The latter complement SVG graphics' formatting capabilities. In contrast to the previous steps, S3 is executed by the rendering engine of the SVG viewer. By using templates for the coarse layout and stylesheets for the personalization of graphical attributes we gain additional flexibility, which allows us to easily adapt the final appearance of process visualizations. This is realized by providing more than one stylesheet for a particular VisModel. The stylesheet to be used is selected lately at runtime, e.g., depending on the organizational unit that requested the process visualization. It is even possible to hide process elements in this late stage using CSS-attributes.

Key to deal with the requirements from Table 1 is the described template mechanism. It enables great flexibility defining the appearance of process elements and also promotes their reuse (Req. 1–3). The look of the resulting process graphs can be customized further using stylesheets (Req. 4). By adopting SVG easy deployment of a visualization component considering available Web-browser plug-ins becomes possible (Req. 5). Further, the availability of frameworks for generating SVG server-side is useful in our context. Thus Req. 5 and 7 are met. In addition to this, SVG allows for the easy definition of process symbols using standard editors (Req. 6). The implementation efforts could be reduced by harking back to existing libraries (e.g. for JavaScript and XPath) (Req. 7).

4 Proof-of-Concept Implementation

We have implemented the described concepts in a powerful proof-of-concept prototype. Fig. 8 depicts sample screens showing the same process in two different execution states and with different appearance. Fig. 8a shows a visualization of the process from Fig. 1, which is similar to what is offered by current process design tools. In Fig. 8b the same process is depicted in another style and with progressed execution state. Reducing the number of elements and streamlining notation often leads to an improved readability as in the given case. This is particularly suitable for monitoring components highlighting activity execution states using different colors. Similarly, another VisModel using colors for identifying participating actors and varying border styles for representing execution states can be defined. Additional information on less important process attributes can be visualized using tool-tips (cf. Fig. 8b). Since we apply SVG, processes can be monitored with standard Web browsers. Usually they include respective plugins providing basic operations (e.g., zooming) by default. Advanced SVG features like animation and scripting have enabled us to interact with users in a sophisticated way, e.g., by replaying process execution with animated state transitions. Our demonstrator is implemented using Java and other standard techniques like XML, XPath, CSS, and JavaScript.



setup test
State: running
Start date: 28-02-2006
Responsible: Mr. Lucas

Fig. 8. Screenshots showing two different presentations of the same process

5 Related Work

There are numerous Workflow Management systems (WfMS) which enable the definition, execution, and monitoring of business processes. In particular, the modeling and monitoring components of these WfMS provide visualization support for model and instance data. However, only those process data can be visualized which are under control of the WfMS (e.g., WBI Monitor [4]). A more open approach is followed by process performance management (PPM) tools (e.g. ARIS PPM [5]), which support the monitoring of processes whose data is scattered over multiple information systems. Altogether these tools show limitations with respect to their visualization component. Neither can the visualization of a process be personalized nor can it be adapted to the current context. In particular, processes are always displayed as drawn by the process designer, and the discussed requirements are not met. Furthermore, at the process instance level visualization support is mainly restricted to the control flow perspective, whereas other process aspects (e.g., data flow, resources, application data, etc.) cannot be displayed. Finally, the Web interfaces of current tools are rather poor and also not adaptable to users' needs. For these reasons, current monitoring tools are mainly used by administrators and developers. A more flexible, but also more complex approach is offered by generic visualization software; e.g., ILOG JViews is not bound to a specific WfMS or workflow meta model and therefore enables more flexible, but also more complex to realize process visualizations.

The area of Information Visualization deals with the use of computer-supported, interactive, visual representations of data to amplify cognition [6]. Several approaches are available for the visualization of general graph structures [7]. However, there is literature dealing with BP visualization [8,9,10,11]. Most approaches focus on special aspects rather than providing a complete picture. Examples include the layouting of certain process graphs [9], the mapping to SVG [10,12], and the adaptation of the set of displayed process elements [13]. ArchiMate [11] is more ambitious and supports different visualizations and viewpoints of enterprise architectures for different user groups. However, most of the discussed requirements have not been addressed by these approaches.

6 Summary and Outlook

We have presented an approach for the personalized visualization of process model and process instance data. For this we have introduced a VisModel that comprises all configuration parameters providing adaptable BP visualizations. Key concept is the flexible definition of process symbols independent from process model data. This has been realized based on a powerful template mechanism. Due to lack of space we have explained basic concepts by means of simple examples rather than providing formal considerations.

The configuration of process visualizations, i.e., the specification of a VisModel, is a complex task that requires writing XML-code. We plan to build a sophisticated tool that allows for the graphical definition of a VisModel as well as for template reuse. Template generation will be facilitated integrating an SVG editor. Layouting general process graphs is another complex task [14]. This and other challenges have been factored out in this paper. At the moment we opt for using existing positioning information of process elements, but we aim at replacing this workaround with sophisticated layout algorithms. Layouting will be introduced to the formatting task from Fig. 4 (after Step S2), where the resulting graph objects can be used to calculate the adequate layout. Automatic layout even gets more important when taking into account more advanced issues like view mechanisms or different visualization forms of process data (e.g., swim lanes, gantt charts, etc.). Finally, in Proviado several other activities are on their way to accomplish all tasks depicted in Fig. 4. This includes view generation (S0), access control, and process data integration.

References

1. Havey, M.: Essential Business Process Modeling. O'Reilly Media (2005)
2. Bobrik, R., Reichert, M., Bauer, T.: Requirements for the Visualization of System-Spanning Business Processes. In: Proc. 16th Int. Workshop on Database and Expert Systems Applications (DEXA), Copenhagen, Denmark (2005) 948–954
3. Jolif, C.: Comparison between XML to SVG Transformation Mechanisms. In: Proc. SVG Open'05, Enschede, Netherlands (2005)
4. IBM: IBM WBI Monitor V. 4.2.3. (2003) IBM Report.
5. IDS Scheer AG: ARIS Process Performance Manager (PPM). White Paper (2004)
6. Card, S.K., MacKinlay, J.D., Shneiderman, B.: Readings in Information Visualization: Using Vision to Think. Morgan Kaufmann, New York (1999)
7. Herman, I., Melançon, G., Marshall, M.S.: Graph Visualization and Navigation in Information Visualization: A Survey. IEEE Transactions on Visualization and Computer Graphics **6** (2000) 24–43
8. Luttighuis, P.O., Lankhorst, M., van de Wetering, R., Bal, R., van den Berg, H.: Visualising Business Processes. Computer Languages **27** (2001)
9. Six, J.M., Tollis, I.G.: Automated Visualization of Process Diagrams. In: Proc. 9th Int. Symp. on Graph Drawing (GD '01), Vienna, Austria (2002)
10. Koolwaaij, J.W., Fennema, P., van Leeuwen, D.: SVG for Process Visualization. In: SVG Open 2003, Vancouver (2003)
11. Steen, M., Akehurst, D., ter Doest, H., Lankhorst, M.: Supporting Viewpoint-Oriented Enterprise Architecture. In: 8th Int. Enterprise Dist. Object Computing Conf. (EDOC), Monterey, California (2004) 201–211
12. Mendling, J., Brabenetz, A., Neumann, G.: EPML2SVG - Generating Websites from EPML Processes. In: Proc. of the 3rd GI Workshop on Event-Driven Process Chains (EPK 2004), Luxembourg (2004)
13. Streit, A., Pham, B., Brown, R.: Visualization support for managing large business process specifications. In: Proc. 3rd Int. Conf. Business Process Management (BPM). Volume 3649 of LNCS., Nancy, France (2005) 205–219
14. Rinderle, S., Bobrik, R., Reichert, M., Bauer, T.: Business process visualization - use cases, challenges, solutions. In: 8th International Conference on Enterprise Information Systems (ICEIS'06), Paphos, Cyprus (2006)

Service-Oriented Data and Process Models for Personalization and Collaboration in e-Business

Chien-Chih Yu

National ChengChi University, Taipei, Taiwan, ROC
ccyu@mis.nccu.edu.tw

Abstract. Providing personalized and collaborative services on the web is critical for creating customer and business values in many e-business domains such as e-shop, e-marketplace, e-news, e-learning, e-finance, and e-tourism. The goal of this paper is to propose a generic service-oriented framework and modeling techniques for facilitating the development of a powerful personalized and collaborative e-service system that is adaptable for use in various e-business applications. A unified data model as well as integrated process models for supporting advanced e-service requirements including search, recommendation, customization, collaboration, negotiation, and trust management are presented and discussed with examples.

1 Introduction

The increasing availability of web technologies and high accessibility of e-commerce (EC)/e-business (EB) applications have strongly pushed enterprises to adopt innovative business models and processes for leveraging their organizational capabilities and market competitiveness. Emerging market-oriented EC/EB business models include e-Shop, e-Mall, e-Procurement, e-Marketplace, e-Auction, Virtual Communities, Value Chain Service Provider, Value Chain Integrator, Collaboration Platforms, Information Intermediaries, and Trust Service Provider etc [14,21]. On the other hand, in addition to transaction and payment process, product/supplier search and discovery, recommendation and selection, auction and negotiation etc have been indicated as major service-oriented processes for delivering advanced business services to create efficiency and effectiveness in serving customers [6,16,20]. For the past few years, many web-based application-level e-services have already been provided in various EB domains, for instances, on-demand news search, subscription and delivery in the e-news sector, on-line course selection and registration in the e-learning sector, on-line investment portfolio recommendation and management in the e-finance sector, on-line airline reservation and package tour recommendation in the e-tourism sector, as well as on-line matching, auction, and negotiation in e-shop and e-marketplace sectors. Among many e-service characteristics and functions, personalization and customization as well as collaboration and communication have frequently been pointed out as critical factors for attracting and retaining individual and group customers, as well as for increasing market shares and generating more up-sell/cross-sell revenues [1,2,17,18,23,24].

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 72 – 81, 2006.

Personalization is often referred as the ability to provide content and services tailored to individual customer's needs based on his preferences and behavior, or on the profiles of others with similar actions [1,24]. Besides the personalization, collaboration has also been considered as the most important ingredient of EC/EB applications and is often seen as the stimulus and mechanism for social interactions that offer supports for coordination, communication, and negotiation among group members [2,17]. As web-based technologies and applications advanced and expanded quickly, more information–intensive and decision-oriented services are desired in almost all EB industries to support personalized and group-based customer needs. Consequently, there is no doubt that providing personalized and collaborative e-services on the web to meet individual and group-based customer needs as well as to attain process efficiency and effectiveness have become essential means for creating customer values and for sustaining business profitability and competitiveness. However, previous research works related to e-services and web services as well as web personalization and collaboration areas focus mainly on specific technical subjects such as service platform and middleware, user profile modeling, service modeling and architecture, service metadata development and management, service search and recommendation, service composition and execution, as well as functions and processes for delivering search, recommendation, and negotiation services in various application domains [3,8,10,11,12,13]. As a result, the lack is significant regarding the development of an ontology-based e-service framework that is adaptable to a variety of e-business application domains for effectively specifying and sufficiently representing desired business-oriented personalized and collaborative e-services [4,7,17,24]. Meanwhile, unified business data and process models for facilitating the implementation and operation of these desired inter-related personalized and collaborative e-services remain as critical yet less-touched issues that deserve more in-depth exploration. In this paper, we aim at proposing a service-oriented architecture, a unified data model, as well as integrated process models for supporting the development and delivery of advanced e-services including search, recommendation, customization, coordination, communication, auction, negotiation, and trust management to meet personalized and community-based e-service requirements. The objective is to provide a generalized e-service framework and a system design guideline for building a powerful personalized and collaborative information system that is adaptable for use in various e-service industries. The rest of this paper is organized as follows. A brief review of related works is provided in section 2. The integrated e-service framework is presented is section 3, followed by the presentation of a unified data model and application-level process models in section 4 and section 5 respectively. The final section contains the conclusion and directions of future research.

2 Related Works

In this section, previous research works related to web personalization, web collaboration, and web services technologies are reviewed and the needs for further studies are identified.

2.1 Web Personalization

In recent years, the web personalization issue has been addressed in different application domains with different perspectives. Godoy et al (2004) propose an interface-agent approach for personalizing web-based information search and newspaper generation. Major tasks performed by interface agents include searching and filtering relevant web documents, generating personalized digital newspapers, as well as modeling and adapting user interests into a user profile [11]. Dolog et al (2004) address the demands of more effective personalization functionalities for learning in open environments to provide learner orientation and individualized access support. They propose a service-based architecture for establishing personalized e-learning systems, where personalized search, recommendation, and link generation are described as main personalization services [10]. Harvey et al (2005) report that an effective decision support system must produce valued information while taking into account user's constraints and preferences on resources. They propose a hierarchical multi-agent system that is driven by a user model and uses a negotiation process to solicit and organize agents to produce and assemble information for making buy/don't-buy decisions on a single investment [12]. Werthner and Ricci (2004) indicate new ways for travel and tourism industries to satisfy consumer needs by means of more customized and configured tourism-related products and services. They expect to see tourists specifying their needs and using intermediary services such as personalized recommendation or reverse auction to select products and vendors for fulfilling their demands [22]. Rigou et al (2004) present a k-windows algorithm for efficient personalized clustering in online shopping environment to deliver the list of tailored products based on customer's needs [19]. Collectively, it can be seen that the personalized services identified in various business domains include the directory and search services, the selection and recommendation services, the self-planning and customization services, as well as the auction and negotiation services.

2.2 Web Collaboration

Among researches related to web collaboration, Bafoutsou and Mentzas (2002) define collaborative computing as the use of computers to support coordination and cooperation of two or more people who attempt to perform a task or solve a problem together [2]. They conduct a survey of collaborative tools and classify them into four functional service categories including group file and document handling, computer conferencing, electronic meeting system, and electronic workplace. Considering collaborative commerce as an emerging paradigm in which a variety of business stakeholders collaborate interactively to sustain competitive advantage, Park et al. (2004) propose a role-driven component-oriented methodology for developing collaborative commerce systems [17]. Kim and Gratch (2004) propose a planner-independent collaborative system architecture to support collaborative planning interactions [15]. Divitini et al. (2004) discuss the need for mobile share workspaces as well as an experimental platform for ubiquitous collaboration [9]. The mobile collaboration service platform offers services for creating and maintaining users, activities, and resources, for dynamically configuring these entities, and for maintaining the presence model. Cai (2005) presents a methodology and a framework for modeling the stakeholders' social interactions and improving their collaboration processes [3]. Three key factors of the collaborative management identified are process, perspective, and conflict management. Orriens and Yang (2005), in their attempt to improve composite web service development, introduce a business collaboration design framework which uses a blend of design perspectives, facets, and aspects for effectively developing and delivering business collaborations [17]. In their approach, the design facets focus on the specification of business description elements including what, how, where, when, who, and why. It can be seen that collaborative services have been discussed from varying angles including the collaborative computing, commerce, planning, design, and process, as well as the mobile collaboration perspectives. Both the framework and service functions are diversified.

In summary, previous works on personalization and collaboration usually took a narrow view and focused only on specific functions or techniques. Therefore, there are needs of service-oriented architectures and systems, unified data models, and integrated process models that can fully incorporate and structurally represent all required participating roles, products and services, needs, preferences and constraints, as well as processes and models for guiding enterprises to innovate, create, deliver, and manage suitable personalized and collaborative e-services for their customers.

3 The e-Service Framework

For supporting advanced multi-dimensional personalization and collaboration, we propose a service-oriented framework in which customer-centric e-services include profile creation and management, navigation and search, recommendation and selection, do-it-yourself (DIY) planning and customization, collaboration and communication, auction and negotiation, tracking and feedback control, as well as trust management, in addition to the usual transaction and payment service. Figure 1 depicts the proposed service-oriented framework. Contents and functions of these services are described below.

Profile Creation and Management Services: Services provide in this group allow customers to create and maintain their multi-dimensional personal profiles including basic personal information, interests and preferences, goals and constraints (e.g. return on financial investment and budget limits), location/time and device situations (e.g. on trip with a cell phone), as well as evaluation criteria and weights for selecting products, services, and providers. Also included are facilities for creating personalized web pages along with subject directories, bookmarks, and annotations.

Navigation and Search Services: These services provide customers with navigation and search mechanisms to retrieve and browse product and service information based on subject hierarchy and keywords or full text. etc. Customers can also save the resulting web sites or content pages in a personal favorite list for future use.

Recommendation and Selection Services: As decision support services, the recommender and selection functions use the customer's inserting request in association with his previously stored personal profile to activate the evaluation process using system default or personalized evaluation criteria. The result of the recommendation services shows system recommended products, services, and vendors that match the customer's needs and preferences with the highest degree. Collaborative filtering and case-based reasoning (CBR) techniques are also used for deriving recommendations. Customers can change their requests and compare the recommended results to select final products/services and put them in a shopping cart.

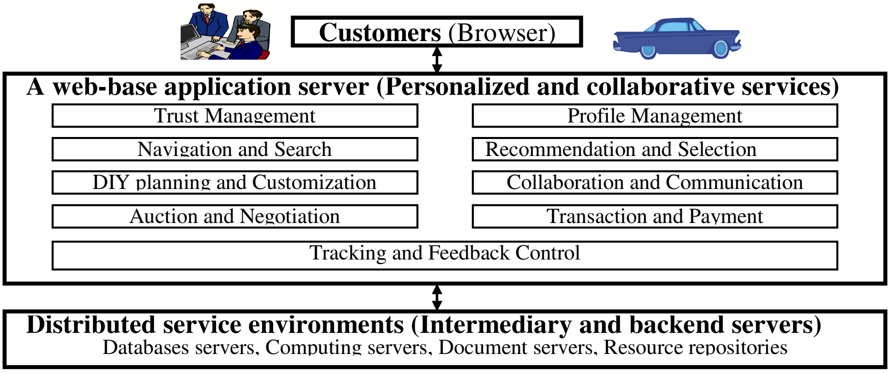

Customers (Browser)
A web-base application server (Personalized and collaborative services)
Trust Management
Profile Management
Navigation and Search
Recommendation and Selection
DIY planning and Customization
Collaboration and Communication
Auction and Negotiation
Transaction and Payment
Tracking and Feedback Control
Distributed service environments (Intermediary and backend servers)
Databases servers, Computing servers, Document servers, Resource repositories

Fig. 1. A service-oriented framework for personalized and collaborative services

DIY Planning and Customization Services: Personalized planning and customized product design are main functions of this services group. Through interactive steps, the customer can specify product and service specifications, and then form a personalized purchase plan by bundling desired products and services. Some examples of customized products include DIY investment portfolios, travel plans, and course materials in e-finance, e-tourism, and e-learning domains respectively.

Collaboration and Communication Services: The basic collaboration services allow customers to form special interest groups and virtual communities, to set up community forums and communication channels, as well as to collaboratively work on specific subjects with others for making group-oriented decisions. Network conferencing, brainstorming, voting are associated mechanisms for group decision making. Communication services include email, bulletin boards, and online chatting.

Auction and Negotiation Services: This group of services provides a dynamic and competitive pricing environment for customers to hold better bargaining positions. The auction services allow customers to issue product/service requests with specified needs and terms, and to launch reverse auction sessions that call for product/service providers to bid on the posted individual or group purchase plans. The submitted bids are evaluated according to pre-specified criteria. Providers with satisfactory cost/benefit levels are selected as candidates for contract negotiation. The negotiation services allow customers to negotiate terms and contracts with chosen providers.

Tracking and Feedback Control Services: This service group provides mechanisms via fixed and mobile devices for customers to track in-progress order transactions, to request for technical support, as well as to make necessary changes for controlling the qualities of products and services. Customers' satisfaction levels about the products and services are collected using feedback control facilities. Also supported are FAQ and complaint handling functions.

Trust Management Services: Provided by independent intermediary service providers, trust management services allow both customers and product/service providers to register and attain authorized trust seals and certificates. The trust certificates are accessible on line for ensuring secure transactions between customers and product/service providers.

Using web browsers, consumers can access application-level personalized and collaborative services from application servers provided by direct product/service providers and/or intermediary service providers. By analyzing customer's inputs and profile, application servers of specific domains such as e-tourism and e-learning systems activate proper service processes and establish links to necessary intermediary and backend web service servers for carrying out all required retrieval, computational and composing activities to deliver customer-requested services.

4 The Unified Data Model

The main information categories needed for sufficiently supporting web-based personalization and collaboration include customers, products and services, product and service providers, as well as functional processes and models.

Customers: In the customers category, multi-dimensional customer profiles are created to specify data objects including customers' basic information, needs and preferences, constraints and time/location situations. The basic information object contains personal data elements such as name, gender, age, phone number, and email, etc. The needs and preferences objects contain data elements that specify customers' needs and preferences related to specific products and services. Taking the e-tourism application as an example, the needs and preferences data for package tour selection include countries and cities, specific destinations and sightseeing points, specific hotels and restaurants, range of departure dates etc, as well as preferred destination features such as SPA, cultural heritages, amusement parks, art museums etc, and preferred hotel features such as ranks, locations, and facilities, and so on. The constraints and time/location objects contain data elements that specify customers' buying constraints in acquiring the products and services, as well as the device types, and time/location conditions for executing transactional activities. For instance, the customer's USD 1000 budget limit for a package tour and his current situation of being on a car with a PDA, are specified constraint and situation data for processing the specific e-tourism applications.

Products and Services: In the product category, multi-dimensional product profiles that specify products' basic information and detail specifications are created. The basic product information objects contain first-level data elements that describe products in a compact way as shown in a product catalog. Major data elements include product ID, product name, product type, price, and manufacturer. The detail product specification objects provide data elements for completely specifying the products. Using the aforementioned package tour as a product example, the second-level product specification data elements include countries, cities, destinations, sightseeing points, hotels, restaurants, departure dates, as well as specific features about destinations, hotels, and transportations etc. Similarly, basic information and detail specification objects and associated data elements of the service category can be defined in the same way. In addition, ID or name data of processes and models needed for executing a specific service are also specified in this category.

Product and Service Providers: In the product and service providers category, the basic provider information object containing data elements such as provider ID, name, type, phone number, email, and web site, as well as the product/service offerings object containing data elements such as product class, product ID, product name, and list price, etc are both specified.

Processes and Models: In the functional processes and models category, the process object related to a specific service describes the service process in terms of process ID, process name, service functions/activities, associated decision/knowledge models, and process input/output files and elements; while the model object specifies computational models by using model ID, model name, model input/output files and elements, as well as computing software and programs. For an example, the recommendation service for recommending personalized package tour involves a package tour retrieval process and an evaluation process. The package evaluation process uses a multi-criteria evaluation model and a rule model for scoring, evaluating, and ranking providers' package tours, and then presents package tours with the higher matching scores as recommended candidates for customers' inspection and selection. Figure 2 shows an object-oriented (OO) data model with these identified service objects. If a relational DBMS is used for database implementation, then the OO conceptual data model must be mapped into an internal relational data model.

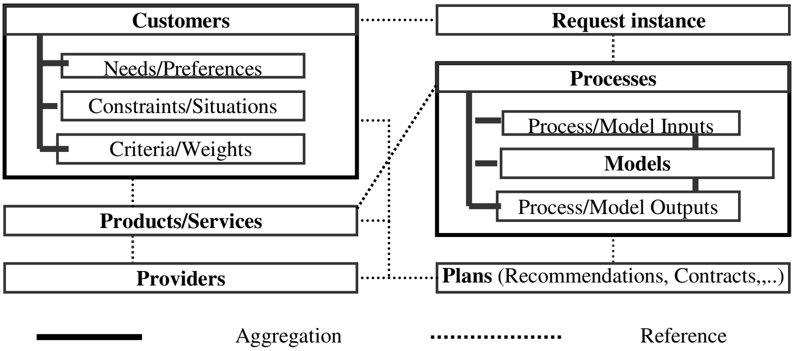


Fig. 2. An OO conceptual data model for personalized and collaborative services

For a specific request instance in the package tour selection example, the customer specifies his needs and preferences of package tours by inputting values and weights for the chosen data elements to be used in the evaluation process. The total matching scores of package tours provided by different travel agencies can be generated by first retrieving selected data elements and then triggering the specified package tour evaluation model and rules. Package tours with final scores exceeding the customer's pre-specified satisfaction level are presented as recommendations to the customer. For the DIY planning and customization services, information objects of customers and products, and a self-design process are involved. For activating the collaboration services, multiple customers, multiple alternatives for a product, as well as a plan proposing process, an idea generation and exchange process, a voting process, and a finalized group plan presentation process are involved. For a reverse auction service, involving information objects include the customers, products, providers, and an auction initiating process, a bidding process, and a bid evaluation and selection process. As for processing the negotiation services, the customers, the products, the providers, as well as a proposal exchange process, a proposal evaluation process, and a contract generation process are required. Therefore, the proposed unified data model is capable of supporting all required personalized and collaborative services.

5 The Integrated Process Model

Implementing a web information system with integrated personalized and collaborative services in various domains such as e-tourism and e-finance, customers such as travelers and investors can easily perform the following service processes.

1. To register and obtain an authorized certificate as a reliable customer for accessing services and conducting secure booking and payment transactions.
2. To create and maintain personal profiles for specifying personalized needs, preferences, constraints, and evaluation criteria.
3. To navigate, search, and browse products and services information that are relevant to the selected subjects or specified search criteria by using desktop computers or mobile devices, as well as to create lists of favorite resource links.
4. To activate the recommendation and selection process using direct-input or pre-specified needs, preferences, evaluation criteria and weights for receiving system recommended products and services, and for choosing products and providers that match the needs and preferences with the highest satisfaction level.
5. To design customized products/services when no existing ones responded to the search and recommendation processes meet the customer's needs and preferences.
6. To locate and organize customers of similar interests in a community to exchange ideas and collaboratively design and develop group-based products and services.
7. To propose products and services requests and initiate reverse-auction sessions for selecting providers with the best bids as candidates for negotiation.
8. To negotiate contracts and terms with chosen product/service providers and develop the final contracts for implementation.
9. To place orders and issue payments for selected/contracted products and services.
10. To track the progress of transactions, access technical supports, and give feedbacks related to the use of products and services, as well as to dynamically make necessary changes on needs and preferences for controlling the qualities of products and services during the operational stage.

Figure 3 illustrates an integrated process model for streamlining these personalized and collaborative services. With the proposed integrated personalization and customization services and processes, all phases of customers' product/service search and decision-making processes can be fully supported.

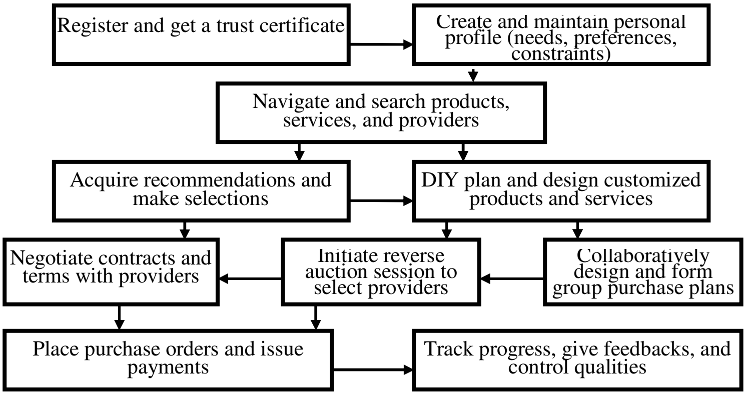

Register and get a trust certificate
Create and maintain personal profile (needs, preferences, constraints)
Navigate and search products, services, and providers
Acquire recommendations and make selections
DIY plan and design customized products and services
Negotiate contracts and terms with providers
Initiate reverse auction session to select providers
Collaboratively design and form group purchase plans
Place purchase orders and issue payments
Track progress, give feedbacks, and control qualities

Fig. 3. The integrated process model of personalized and collaborative services

6 Conclusion

In this paper, we accomplish the objectives of (1) addressing the need of personalized and collaborative services in a variety of business sectors, (2) providing a compact literature review on related works, (3) proposing a service-oriented framework for integrating the desired personalized, collaborative, and supportive services, (4) presenting an unified conceptual data model and an integrated process model for efficiently and effectively developing a web-based information system to incorporate all advanced personalized and collaborative services. Furthermore, the service-oriented framework, data model and process model are adaptable to various application domains for facilitating the implementation and operation of personalized and collaborative services. Future research topics include practically implementing the proposed service framework, data model and process model to specific application domains, as well as measuring system and services performances from both customer and business perspectives. Also to be explored is the issue of extending web service related description, semantic web ontology, and execution languages such as WSDL, OWL-S, and BPEL to standardized the business–level service framework and models for working with the SOAP-based web service environments.

References

1. Adomavicius, G. Tuzhilin, A.: Personalization Technologies: A Process-Oriented Perspective. Communications of the ACM. 48(10) (2005) 83-90.
2. Bafoutsou, G. and Mentzas, G.: Review and Functional Classification of Collaborative Systems. International Journal of Information Management. 22(4) (2002) 281-305.
3. Cai, J.: A Social Interaction Analysis Methodology for Improving E-Collaboration Over the Internet. Electronic Commerce Research and Applications. 4(2) (2005) 85-99.
4. Casati, F. et al. Business-Oriented Management of Web Services. Communications of the ACM. 46(10) (2003) 55-60.

5. Chen, C.M., Lee, H.M., and Chen, Y.H.: Personalized e-Learning System Using Item Response Theory. Computers & Education. 44(3) (2005) 237-255.
6. Choi, H.R. et al.: Implementation of Framework for Developing Multi-agent Based Automated Negotiation Systems, In Proceedings of the 7th International Conference on Electronic Commerce, (2005) 306-315.
7. Crawford, C.H. et al.: Toward an On Demand Service-Oriented Architecture. IBM System Journal. 44(1) (2005) 81-107.
8. Dikaiakos, M. D. and Zeinalipour-Yazti, D.: A Distributed Middleware Infrastructure for Personalized Services. Computer Communications. 27(15) (2004) 1464-1480.
9. Divitini, M., Farshchain, B. A., and Samset, H.: UbiCollab: Collaboration Support for Mobile Users, In Proceedings of the 2004 ACM Symposium on Applied Computing, (2004) 1191-1195.
10. Dolog, P., Henze, N., Nejdl, W., and Sintek, M.: Personalization in Distributed e-Learning Environment, In Proceedings of the 13th international World Wide Web Conference, (2004) 170-179.
11. Godoy, D., Shiaffino, S., and Amandi, A.: Interface Agents Personalizing Web-Based Tasks. Cognitive Systems Research. 5(3) (2004) 207-222.
12. Harvey, T., Decker, K., and Carberry, S.: Multi-Agent Decision Support via User-Modeling, In Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems, (2005) 222-229.
13. Hull, R. and Su, J.: Tools for Composite Web Services: A Short Overview. SIGMOD Record. 34(2) (2005) 86-95.
14. Jones, S. et al.: Trust Requirements in e-Business. Communications of the ACM. 43(12) (2000) 81-87.
15. Kim, W. and Gratch, J.: A Planner-Independent Collaborative Planning Assistant, In Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems, (2004) 764-771.
16. Liu, D.R., Shen, M., and Liao, C.T.: Designing a Composite e-Service Platform with Recommendation Function. Computer Standards and Interfaces. 25(2) (2003) 103-117.
17. Orriens, B. and Yang, J.: Establishing and Maintaining Compatibility in Service Oriented Business Collaboration, In Proceedings of the 7th International Conference on Electronic Commerce, (2005) 446-453.
18. Park, H., Suh, W., and Lee, H.: A Role-Driven Component-Oriented Methodology for Developing Collaborative Commerce Systems. Information and Software Technology. 46(12) (2004) 819-837.
19. Rigou, M., Sirmakessis, S., and Tsakalidis, A.: A Computational Geometry Approach to Web Personalization, In Proceedings of the IEEE International Conference on E-Commerce Technology, (2004) 4ps.
20. Schilke, S.W., Bleimann, U., Furnell, S.M., and Phippen, A.D.: Multi-Dimensional-Personalisation for Location and Interest-Based Recommendation. Internet Research. 14(5) (2004) 379-385.
21. Timmers, P.: Business Models for Electronic Markets. Electronic Markets. 8(2) (1998) 3-8.
22. Werthner, H. and Ricci, F.: E-Commerce and Tourism. Communications of the ACM. 47(12) (2004) 101-105.
23. Yu, C.C.: Personalized and Community Decision Support in eTourism Intermediaries. Lecture Notes in Computer Science, Vol. 3588, Springer-Verlag, (2005) 900-909.
24. Zhang, D., Chen, M., and Zhou, L.: Dynamic and Personalized Web Services Composition in E-Business. Information Systems Management. 22(3) (2005) 50-65.

A Framework for Raising Collaboration Levels on the Internet

Igor Hawryszkiewycz

Department of Information systems
University of Technology, Sydney
igorh@it.uts.edu.au

Abstract. People in most business processes still use the basic technologies such as e-mail or intranet information portals to collaborate on the Internet. Collaboration is thus primarily restricted to sharing of documents and comments between individuals or within small groups. The paper examines ways to provide higher levels of support for collaboration especially for collaboration in the large. The paper first defines levels of collaboration and technologies available to support each level. It then develops a definition of collaboration capability in terms of these levels and proposes methods that use collaboration capability levels to define collaboration requirements for particular applications. It then provides ways to extend workspaces to support collaboration in the large.

Keywords: Situation analysis, Collaboration, Knowledge Sharing.

1 Introduction

Collaboration and use of technology are now generally recognized as necessary to improve business processes. For example collaborative systems were effective in reducing the preparation of response documents from 4 to 2 months in consulting organizations [1]. Another example is the capture of best practices to share within organizations [2]. Such benefits are particularly possible in processes that are knowledge intensive [3]. This particularly applies where people are required to deal with increasingly complex situations that require a quick response. One obvious example here is emergency response systems [4] that must quickly respond to rapidly emerging situations. In business processes there are also situations that require response. Examples can be falling market share, a new competitor or opportunity provided by a new technology. The general consensus is that organizations must become agile and quickly respond to such situations in creative and innovative ways, a process sometimes called *situation analysis*, which requires ways to support people within the organization to collaborate to share their knowledge to quickly provide innovative solutions.

In most users the only collaborative technologies used are e-mail, and portals for collaboration. However, research [5] has shown that such support is generally restricted to exchange of information as this often requires users to themselves spend time in organizing e-mail into folders recording relationships between messages and ensuring that their information is up to date [6]. Collaboration in the large poses

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 82 – 91, 2006.

further problems as it must be supported between functional units as well as within functional units and requires planning of work processes as well as coordination of these processes.

Often the choice of collaborative tools is left to individuals whose goal is to solve their local problem leading to complications further down the line. This paper describes an organizational approach to the design of collaborative systems. It first proposes a measure of collaboration that measure ways to obtain value across the whole organization. It then shows how this framework can be used both to identify problems in current systems as well as to design new systems and ways to select required technologies. The framework identifies levels of collaboration and technical strategies for realizing them. The paper begins by describing the framework.

2 The Framework

The framework for collaboration proposed in this paper includes three dimensions, namely:

Level of collaboration, which define a basic measure of collaboration in the activities, and

Collaboration capability that defines the contribution of collaboration to the business value of the organization.

Defining a set of collaboration levels provides a way to measure the effectiveness of collaboration in the organization. This in turn is used to evaluate the collaboration capability, which provides a way to identify collaboration shortfalls. Collaboration capability can be used to define *collaboration requirements* for a given situation. The paper then provides guidelines for choosing technologies to satisfy the collaboration requirements. Usually this will require raising collaborative levels. This can be applied across the whole organization or refined to apply to the different activities found in situation management.

The framework recognizes that there are a large number of activities when dealing with a situation. Each activity may need different kinds of collaborative support. The framework can be used to identify critical activities and raise collaboration significantly in selected activities, as for example speeding up a response to new business opportunities [7]. On the other hand as for example in the case of collaboration in business process such as supply chain management it can raise the level gradually to add value across the whole process.

2.1 Levels of Collaboration

The collaboration levels proposed in this paper are shown in Table 1.

Table 1 describes each level and its characteristic, as well as the knowledge needed to realize the level. Five levels are defined, namely:

- Event notification, where roles are informed of any changes that effect the roles.
- Document sharing, where documents are distributed between responsible roles,
- Work process support, which often defines monitoring levels of activity and sending reminders to collaborators,

- Joint work, where users work together in a synchronous manner, and
- Joint goal setting, where people jointly decide how they will work together.

Table 1. Levels of Collaboration

Level	Characteristics of collaboration levels	Knowledge requirements
Collaboration level 1- Event Notification	Informing people about events related to their roles. Presenting the functional situation globally.	People responsibilities in the organization and their location or contact.
Collaboration level 2 - Document Sharing	Sharing explicit information. Presenting latest information to roles responsible for functional units. Obtaining comments on information.	Role responsibilities and information and documents that they require to effectively carry out their roles. Knowledge of related documents.
Collaboration level 3 - Work process support	Explicit definition of relationships and responsibilities. Definition of relationships between tasks. Group meetings to resolve issues.	Location of experts. Ways to assign responsibilities.
Collaboration level 4 - Joint work	Jointly create and develop artifacts.	Location and responsibilities of people involved in a task.
Collaboration level 5 - Joint planning	Developing shared plans and devising coordination strategies between functional units. Developing and agreeing on work processes.	Optimum team structures for identified situations. Organizational strategy and mission. Responsibilities of different organizational units.

The collaboration levels provide a way to introduce technology with increasing levels of sophistication. Usually collaboration starts by simply notifying people of changes that can impact on their work. Then document sharing is added to ensure that people are provided with information needed to carry out their responsibilities. Subsequent levels are more complex as they must support intense interaction to coordinate activities. Work process support, requires a precise definition of the way a collaborative process takes place and define the responsibilities of identified roles. For example a response to a customer request may define the expertise needed to define a solution, the risk assessment, budgetary evaluation, legal aspects and so on. Joint work is an extension of level 3 by providing ways to carry out synchronously thus reducing completion time. Defining the processes to be followed also requires collaboration and agreement on the ways people will work to achieve organizational goals. This process must be clearly defined and clearly understood and followed. Joint planning requires involved units together plan and agree on their work processes. This level often requires support for asynchronous work as goal setting often includes resolving many imprecisely defined alternatives.

2.2 Collaboration Capability

The idea of collaboration capability comes from earlier adoption of the idea of capability maturity model adopted in software engineering. This centered on defining the process requirements that ensure the development of software products. Similarly the idea of collaboration capability is to define processes needed to ensure the effective sharing of knowledge to develop ways to respond to situations. Table 2 defines the collaboration capability levels.

Table 2. Collaboration Capability Levels

Collaboration Capability	Description	Expected benefits
Capability Level 0 Ad-hoc	Use of current technologies in ad-hoc ways. Messages sent and documents exchanged in unpredicted ways depending on user preferences.	Available technologies such as e-mail or portals facilitate exchange of information.
Capability Level 1 Task coordination.	Policies exist for maintaining awareness in tasks and for sharing documents. Usually requires level 1 and 2 levels of collaboration. Can apply to individual business units or across business units.	Consistency of produced documents and less duplication thus reducing unnecessary work. Improved awareness of what is going on in teams.
Capability Level 2 Process coordination.	Work across different units is coordinated and task progress reported. Requires collaboration level 3.	People across the organization can respond quickly to changes thus reducing completion times. Ability to provide global responses quickly and to capture best practices.
Capability Level 3 Work alignment	Work processes shared across business units. Individuals discuss (usually synchronously) ways documents and processes should be managed before implementing them. Requires collaboration level 4 and higher.	Quicker alignment of business units to business goal. Ability to provide global solutions quicker.

3 Technology Levels

A large number of technologies are available to support collaborative activities. Often different technologies are needed in each activity. For example in identifying a situation the emphasis is on collecting information from many disparate sources and bringing it in suitable form to decision makers. Deciding on a course of action requires technologies to involve experts that use this information to assess the situation and decide on a response. Action execution requires technologies that assist the coordination of tasks involved in the execution. In rapidly evolving situation, such as for example occur in emergencies, the activities are continually evolving and must be closely integrated. Consequently technologies at each activity must also be tightly integrated. Collaboration technologies are classified into the levels shown in Table 3, 4.

Table 3. Available Technologies

State of Technology	**Available Technologies**	**Use of Technologies**
Current Practice	e-mail, information portals, SMS, discussion boards, workflow management systems	Limited especially where large numbers of people are involved [5]. Information portals, which provide access to shared files. These can be useful in individual activities, especially in raising awareness to a situation and collating information. Less common are discussion forums, which allow issues to be discussed and calendars, which provide information about availability of people.
Emerging state of art	Shared whiteboards, video and audio conferencing, screen sharing, instant messaging, blogs and wikis	The state of art technologies emphasize the synchronous aspects of collaboration but usually emphasize individual tasks. They can; • lead to quicker decisions through access to experts and assembly of relevant information, • result in quicker delivery times and knowledge sharing as such exchanges take place in real time, • improve work process support if special roles are created to coordinate joint work. Thus they partially go towards improving capability level 2 but in general are not effective for doing this for large scale collaboration.
Leading edge	Workspaces, virtual co-location, graphical displays	Leading edge technologies offer the best approach to reach higher collaboration capability levels for large scale collaboration. They include workspaces that present the status of all activities and enable users to make decisions taking into account the whole situation. A workspace can present the current state in each activity using an integrated interface enabling quicker global response.

3.1 Matching Technology to Collaboration Level

Table 4 illustrates the applicability of the technologies to the different levels of collaboration.

Table 4. Matching Technology to Levels of Collaboration

Level	Technologies needed	Extending to collaboration in the large
Collaboration level 1 - Event Notification	e-mail alerts, SMS messages. Visual displays. Information portals.	There are no special ways here except mainly to rely on point to point communication.
Collaboration level 2 - Document Sharing	e-mail, web portals, discussion systems, blogs. Some state of art technology needed to improve	e-mail usually insufficient Requires knowledge of relationships between documents
Collaboration level 3 - Work process support	Coordination and Workflow tools, Video displays, Calendar systems. Workflow systems can support predefined processes.	Clear identification of roles and their coordination responsibilities. Multi window displays to display relationships between tasks. Ways to maintain links between different units for operational support. Generally requires leading edge technology.
Collaboration level 4 - Joint work	Shared whiteboards. Video display .Requires state of art technologies.	Identifying smaller tasks and their relationships. Needs advanced coordination tools.
Collaboration level 5 - Joint planning	Synchronous video communication.	Ways to maintain links between different units for decision support.

From Table 4 one can deduce that technologies used in current practice can at best support capability level 1. To do this however will still require careful setting of notification schemes and automatic transmission of documents using notification schemes. To do so requires going beyond simply expecting people to keep track of what is going on. Thus some active ways have to be implemented to inform people of changes and what they need to do. Most communication here also takes place asynchronously.

Some general comments are that with ‘collaboration in the large’ internet technologies will need to support both synchronous and asynchronous work and integrate these with support for work processes. Some requirements are defined in the next section.

4 Designing for Higher Collaboration Capability

Collaboration in the large will push the trend to leading edge technologies in particular workspaces, which will need to support both synchronous and asynchronous work. We have developed a metamodel [8] that defines all these requirements and developed a workspace system, Livenet [9], the implements them. We use this to first define the activities and then specify collaborative requirements of each activity. Figure 1 shows an activity diagram in a typical service organization, which provides a service to a client by packaging other available services. Figure 1 also shows the roles and artifacts that are used in the system. Briefly it includes four activities, namely:

- Packaging the product, which includes identifying client requirements, and ways to meet them by combining existing services or products,
- The required services are purchased from their suppliers,
- Delivering the packaged product and any associated training, and
- Planning and facilitating the process.

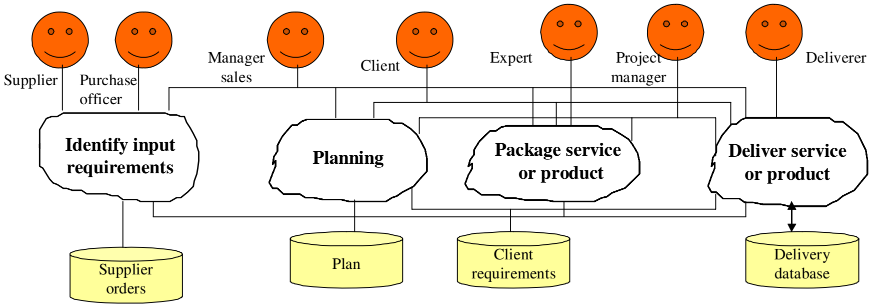


Fig. 1. An Activity Diagram for Service Packaging

The next step is to define the requirements in terms of the business value expected from collaboration. The collaboration level needed to get that value is then identified and technology chosen. In this case the goal is to achieve collaboration capability level 2 to expedite the time from customer request to delivery. This can then be reduced to individual activities as:

Identify input requirements – find the suppliers as quickly as possible,
Packaging service – reduce time to package the product,
Delivery – reduce time to deliver the product and train users.

In the implementation each activity is implemented as a workspace that includes the technologies needed to realize the required collaborative level. Table 4 provides the guidelines for choosing such technologies. These will include lightweight workflow to keep track of progress together with notification schemes and perhaps blogs, or discussion boards, to share experiences and best practices.

Furthermore, integration requires that:

Artifacts must be shared across workspaces with a permission system to clearly define responsibilities for them.
Notifications can be sent across workspaces,

A workspace network is illustrated in Figure 2. Usually there is a central workspace, in this case a business process that links to all services. Each service has its own workspace for each of its activities. It is easy to move between workspace.

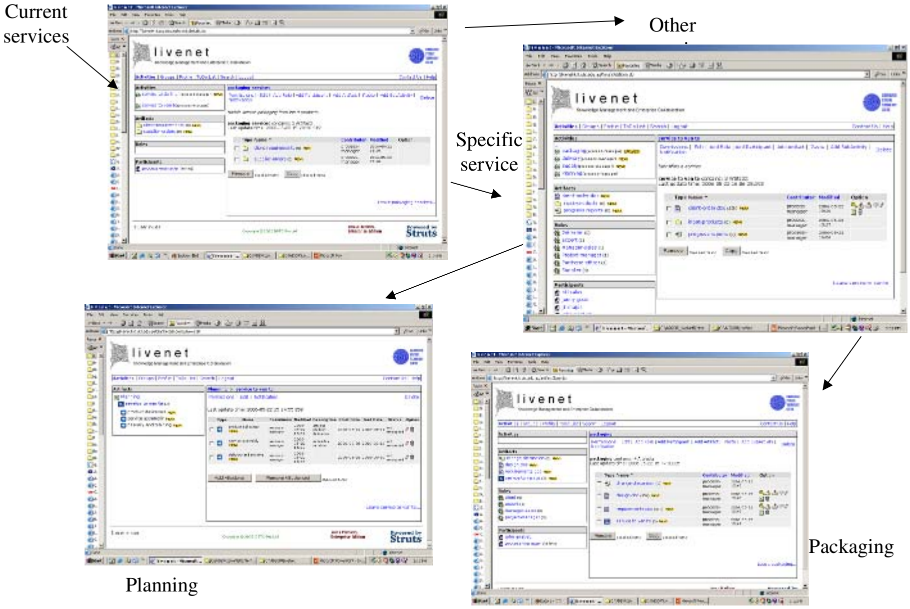
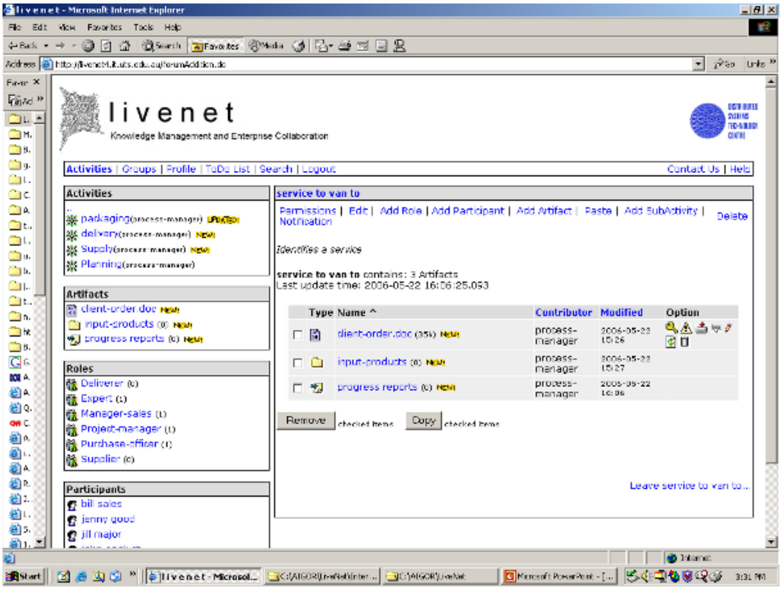


Fig. 2. A workspace Network

For collaboration in the large the workspaces must be integrated. Integration between the workspaces is achieved in a number of ways and usually requires objects to be shared between workspaces and ways to navigate between the workspaces. Typical features include:

- Artifacts can be shared between workspaces with permissions depending on the workspace by a copy and link process.
- People can appear in more than one workspace with different abilities in each workspace,
- Events in one workspace can trigger actions in another.

A single workspace is illustrated in Figure 2. It shows all the objects in the workspace as well as links to related activities and groups.


livenet - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://livenet4.it.uts.edu.au/forumAddition.do
Go Links
livenet
Knowledge Management and Enterprise Collaboration
DISTRIBUTED SYSTEMS TECHNOLOGY CENTRE
Activities | Groups | Profile | ToDo List | Search | Logout
Contact Us | Help
Start
livenet - Microsof...
C:\AIGOR\LiveNet\Inter...
C:\AIGOR\LiveNet
Microsoft PowerPoint - [...
Internet
3:31 PM

Activities

..
packaging(process-manager) UPDATED
delivery(process-manager) NEW!
Supply(process-manager) NEW!
Planning(process-manager)

Artifacts

client-order.doc NEW!
input-products (0) NEW!
progress reports (0) NEW!

Roles

Deliverer (0)
Expert (1)
Manager-sales (1)
Project-manager (1)
Purchase-officer (1)
Supplier (0)

Participants

bill sales
jenny good
jill major

service to van to

Permissions | Edit | Add Role | Add Participant | Add Artifact | Paste | Add SubActivity | Notification Delete

Identifies a service

service to van to contains: 3 Artifacts
Last update time: 2006-05-22 16:06:25.093

	Type	Name ^	Contributor	Modified	Option
□		client-order.doc (35k) NEW!	process-manager	2006-05-22 15:26	
□		input-products (0) NEW!	process-manager	2006-05-22 15:27	
□		progress reports (0) NEW!	process-manager	2006-05-22 16:06	

Remove checked items Copy checked items

Leave service to van to...

Fig. 3. A workspace

5 Future Work

Collaboration in the large can prove to be difficult to manage by people using technology itself. Many applications based on general purpose workspaces fail [10] because of the time needed to change the workspace as the work process changes. Our goal is to develop what is defined here as an active workspace. In this case the workspace possesses sufficient knowledge to assist users to adapt it to current situation. We are currently developing an agent architecture [13] to support such message exchange. A typical scenario would be one where for example a delay is encountered in some project task in a supply chain. The implication of this would be evaluated for each subsequent task and individuals in these tasks would be notified with corrective actions suggested. The research 11, 12]. Each agency can support one activity with the agencies exchanging messages to maintain global awareness.

6 Summary

This paper developed a framework for assessing levels of collaboration and improving collaboration in situations that require the collection and analysis of information and using this to respond to the situation. It then illustrates ways to achieve high collaborative capabilities for collaboration in the large.

References

[1] Hansen, M.T., Nohria, N. and Tierney, T. (1999): "Whats your Strategy for Managing Knowledge" Harvard Business Review, March-April, 1999, pp. 106-116.

[2] Artail, H. "Application of KM measures to the impact of a specialized groupware system on corporate productivity and operations" Information and Management, 2006, Elsevier Press.

[3] Grant, R.M. (1996): "Prospering in Dynamically-competitive Environments: Organizational Capability as Knowledge Integration" *Organization Science*, Vol. 7, No. 4, July, 1996, pp. 375-387.

[4] Bammidi, P.; Moore, K.L. (1994): "Emergency management systems: a systems approach" Systems, Man, and Cybernetics, 1994. 'Humans, Information and Technology'., 1994 IEEE International Conference on Volume 2, 2-5 Oct. 1994 Page(s):1565 - 1570 vol.2 Digital Object Identifier 10.1109/ICSMC.1994.400070

[5] Cummings, J.N., Butler, B. and Kraut, R. (2002): "The Quality of OnLine Social Relationships" Communications of the ACM, Vol. 45, No. 1, July, 2002, pp. 103-111.

[6] Ducheneaut, N. and Bellotti, V. (2001): "E-mail as Habitat' Interactions, September-October, 2001, pp. 30-38.

[7] Hawryszkiewycz, I.T. (1996): "Support Services For Business Networking", in *Proceedings IFIP96*, Canberra, eds. E. Altman and N. Terashima, Chapman and Hall, London, ISBN 0-412-75560-2.

[8] Hawryszkiewycz, I.T. (2005): "A Metamodel for Collaborative Systems" Journal of Computer Information Systems, Spring 2005, pp. 131-146.

[9] Livenet – http://livenet4.it.uts.edu.au

[10] Hummel, T., Schoder, D. and Strauss, R.E. (1996): "Why CSCW Applications Fail: Problems in Support of Lateral Cooperation and the Appropriateness of CSCW Applications:" Proceedings of the Annual Meeting of the Northeast Decision Sciences Institute (NEDSI), April 1996, St. Croix, USA.

[11] Carley, K.M., and Gasser, L. (1999): "Computational Organizational Theory" in Chapter 7 "Computational Organization Theory" by KM Carley & L Gasser in "Multiagent Systems" Gerhard Weiss (Ed) MIT Press -- 1999

[12] Rehfelldt, M., Turowski, K. (2000): "Business models for coordinating next generation enterprises" Proceedings of the IEEE Academia/Industry Working Conference on Research Challenges, pp. 163-168, April 27-29, 2000.

[13] Hawryszkiewycz, I.T. and Lin, A.(2003): "Process Knowledge Support for Emergent Processes" Proceedings of the Second IASTED International Conference on Information and Knowledge Management, Scottsdale, Arizona, November, 2003, pp. 83-87.

Designing Volatile Functionality in E-Commerce Web Applications

Gustavo Rossi*, Andres Nieto, Luciano Mengoni, and Liliana Nuño Silva

LIFIA, Facultad de Informática, UNLP, La Plata, Argentina
{gustavo, anieto, lmengoni, lilica}@sol.info.unlp.edu.ar

Abstract. In this paper we present a flexible design approach and a software framework for integrating dynamic and volatile functionality in Web applications, particularly in e-commerce software. We first motivate our work with some examples. We briefly describe our base design platform (the OOHDM design framework). Next, we show how to deal with services that only apply to a particular set of application objects by clearly decoupling these services from the base conceptual and navigation design and by defining the concept of service affinity. We describe an implementation environment that seamlessly extends Apache Struts with the notion of services and service's affinities. Finally, we compare our approach with others' work and present some further research we are pursuing.

1 Introduction

Complex E-Commerce Web applications are hard to build and harder to maintain. While they initially comprise a myriad of diverse functionality, which makes development a nightmare, their evolution tend to follow difficult to characterize patterns; quite often, new services are added and tested with the application's users community to determine whether they will be consolidated as core application services or not. Moreover, there are services which are known to be temporary, i.e. they are incorporated into the application during some time and later discarded, or they are only activated in specific periods of time. In this paper we are interested in the design and implementation of those, so called volatile requirements and the impact they have on the design model and on the application's architecture. We present an original approach to deal with these requirements modularly; by clearly decoupling the design of these application's modules we simplify evolution.

There are many alternatives to deal with this kind of volatile requirements. One possibility is to clutter design models with new extensions. The main problem with this approach is that it involves intrusive editing and therefore it may introduce mistakes as new functionality is added or edited. A second possibility is to consider that volatile functionality does not deserve to be designed (as it is usually temporary) and deal with these changes only at the implementation level. This approach not only is error prone but it also de-synchronizes design documents with the running system, therefore introducing further problems.

[*] Also CONICET.

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 92 – 101, 2006.

Volatile requirements pose a new challenge: how to design and implement them in order to keep the previously described models and the implementation manageable [15,16]. For example suppose we want to support donations (e.g. as in Amazon after South Asian Tsunami in 2004); this functionality arose suddenly, and implied adding some new (fortunately simple) pages and links from the home page. This kind of additions are usually handled in an ad-hoc way (e.g. at the code level), making design documents obsolete.

Keeping design models up to date is not straightforward: Should we clutter the design models with these new navigation units and then undo the additions when the requirement "expires"? How do we deal with those requirements that are not only volatile but moreover they apply only to some specific objects (e.g. not to the complete set of one class's instances)? Should we modify some specific classes? Add new classes in a hierarchy? Add some new behaviors to existing classes? The main risk of not having a good answer to these questions is that the solution will be to patch running code, making further maintenance even harder.

In this paper we describe our model-based approach for dealing with volatile functionality. We describe a simple approach which can be easily incorporated into the design armory of existing methods. It comprises the definition of a Service layer, describing volatile services both in the conceptual and navigational models, and uses the concept of service's affinities as defined in IUHM [8] to bind new services with application objects. Services are more than just plain behaviors, but may encompass complete (conceptual or navigation) models. We also describe an implementation architecture to show the feasibility of our approach and an extension to Apache Struts that supports the architecture. To make the discussion concrete, we describe our ideas in the context of the Object-Oriented Hypermedia Design Method (OOHDM).

The main contributions of this paper are the following:

- We present a design approach for clearly separating volatile functionality, particularly when it involves the definition of new nodes and links in the Web application.
- We show how to integrate this functionality by using the concept of service affinity.
- Finally, we describe an implementation architecture and framework supporting our ideas.

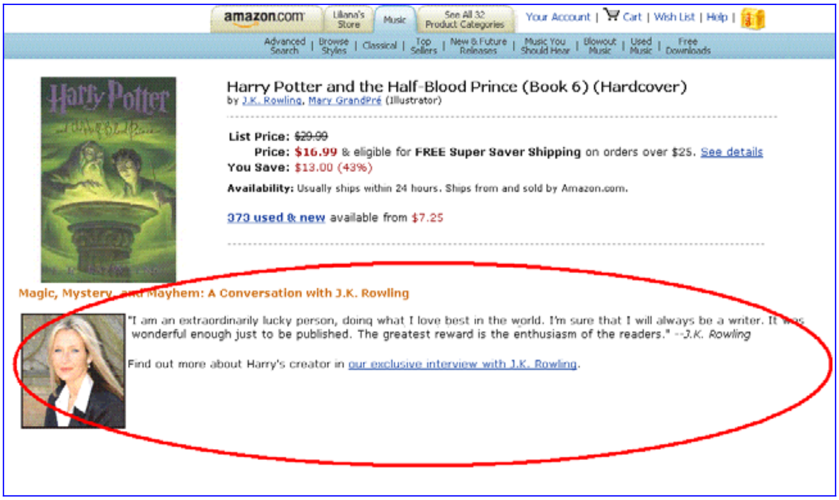
The rest of the paper is organized as follows: In Section 2 we present some simple motivating examples. In Section 3 we describe the core of our approach by discussing services and affinities. In Section 4 we briefly describe and implementation approach. In Section 5 we compare our work with other related approaches and finally in Section 6 we present some concluding remarks and further work on this area.

2 Motivating Examples

In order to show what kind of volatile application functionality we aim to deal with, we next show some examples in the context of the Amazon Web application.

In Figure 1 we show part of the page of the last Harry Potter's book. Below the book information and the editorial reviews, there is an excerpt of an interview with the author, and a link to the full interview which is only accessible from this book and not from others of the series (and certainly it does not make sense in other authors' books). The interview is an aggregation of questions and answers with hypertext links to other books and authors. One can assume that as time passes, this interview (now, a novelty) will be eliminated. We face two problems when designing this simple functionality: how to indicate that it is available from some specific pages, and being volatile, how to keep it separated from the rest of the design.

In Figure 2 we see the page of Rolling Stone's "A bigger Band" CD; in the end of the page (also shown in the Figure) we can see a link to a site for buying tickets for Stones' next concert in Argentina. The same link appears in all Stones' disks. It is reasonable to think that this functionality will be eliminated after the concert is over.


amazon.com
Liliana's Store
Music
See All 32 Product Categories
Your Account | Cart | Wish List | Help |
Advanced Search | Browse Styles | Classical | Top Sellers | New & Future Releases | Music You Should Hear | Blowout Music | Used Music | Free Downloads
Harry Potter

Harry Potter and the Half-Blood Prince (Book 6) (Hardcover)
by J.K. Rowling, Mary GrandPré (Illustrator)

List Price: ~~$29.99~~
Price: $16.99 & eligible for **FREE Super Saver Shipping** on orders over $25. See details
You Save: $13.00 (43%)

Availability: Usually ships within 24 hours. Ships from and sold by Amazon.com.

373 used & new available from $7.25

Magic, Mystery, and Mayhem: A Conversation with J.K. Rowling

"I am an extraordinarily lucky person, doing what I love best in the world. I'm sure that I will always be a writer. It was wonderful enough just to be published. The greatest reward is the enthusiasm of the readers." --*J.K. Rowling*

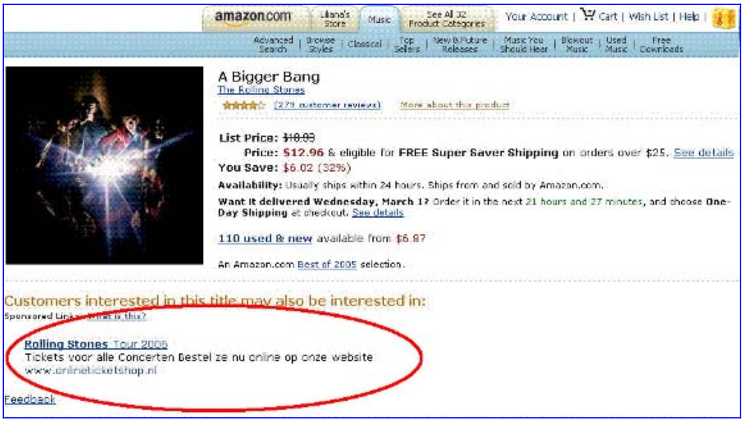
Find out more about Harry's creator in our exclusive interview with J.K. Rowling.

Fig. 1. Interviewing a book's author concert

Similar examples such as the functionality for full search inside a book, the Mozart store (celebrating his 250 anniversary), etc. share the same characteristics: they are known to be volatile and in some cases the new services only apply to some specific pages in the system. A naive approach for solving these problems would be to pollute the design model, by adding the intended information, links or behaviors to the corresponding conceptual and navigational classes. This approach fails because of two main reasons:

- It neglects the fact that certain functionality does not apply to a complete class (e.g., not every book is linked to an interview with the author, not every CD includes a pointer to a ticket selling service)
- It implies that the design models have to be often edited intrusively (e.g. changing attributes and behaviors of a class)

We next elaborate our approach for tackling these problems.


amazon.com
Iliana's Store
Music
See All 32 Product Categories
Your Account | Cart | Wish List | Help
Advanced Search | Browse Styles | Classical | Top Sellers | New & Future Releases | Music You Should Hear | Blowout Music | Used Music | Free Downloads

A Bigger Bang
The Rolling Stones
(279 customer reviews) More about this product

List Price: ~~$18.98~~
Price: $12.96 & eligible for FREE Super Saver Shipping on orders over $25. See details
You Save: $6.02 (32%)
Availability: Usually ships within 24 hours. Ships from and sold by Amazon.com.
Want it delivered Wednesday, March 1? Order it in the next 21 hours and 27 minutes, and choose One-Day Shipping at checkout. See details

110 used & new available from $6.87

An Amazon.com Best of 2005 selection.

Customers interested in this title may also be interested in:
Sponsored Links (What is this?)

Rolling Stones Tour 2006
Tickets voor alle Concerten Bestel ze nu online op onze website
www.onlineticketshop.nl

Feedback

Fig. 2. Selling tickets for a group's concert

3 Our Approach in a Nutshell

The rationale behind our approach is that even the simplest volatile functionality (e.g. the links added to the page in Figure 2) must be modeled and design using good engineering techniques. We think that by surpassing the need to design volatile functionality, we not only compromise the relationships among design models and the actual application but also loose reuse opportunities, as many times a new (volatile) feature might arise once and again in different contexts. A model-based approach, instead, allows increasing the level of abstraction in which we reason with these features, improving comprehension and further evolution. We next present the basic elements of our approach.

3.1 A Brief Description of the OOHDM Model

OOHDM as other development approaches such as OOWS [9], UWE [6] partitions the development space into five activities: requirements gathering, conceptual design, navigation design, abstract interface design and implementation. Though OOHDM does not prescribe a particular strategy for implementing a hypermedia or Web application, its approach can be naturally mapped to object-oriented languages and architectural styles, such as the Model-View-Controller. Some MDA [7] tools already exist for OOHDM [2]; in this paper we describe a semi-automatic approach for generating running implementations which exhibit volatile services.

Usually, new behaviors (or services) are added to corresponding classes, and new node and link classes are incorporated to the existing navigational schema, therefore extending the base navigation topology. As previously indicated, there are two problems with this approach; first it is based on intrusive editing of design models; besides, and as exemplified, there is no easy way to characterize which objects should be the host of new links or services, when they are not defined in the class level. We next describe how we extended the methodology to cope with volatile functionality.

3.2 Modeling Volatile Functionality in OOHDM

Our approach is based on four basic principles:

- We decouple volatile from core functionality: We define two design models; a core model and a model for volatile features (called VService Layer).
- New behaviors, i.e. those which belong to the volatile functionality layer are modeled as first class objects, e.g. following the Command [4] pattern.
- To achieve obliviousness, we use inversion of control, i.e. instead of making core classes aware of their new features, we invert the knowledge relationship. New behaviors know the base classes on top of which they are built.
- We use a separate integration layer to bind core and volatile functionality. In this way, we achieve reusability of core and volatile features and manage irregular extensions.

3.3 The Volatile Services Layer

The introduction of the VService Layer was inspired in part in the IUHM model in which services are described as first class objects. We considered services as a combination and generalization of Commands and Decorators [3]. A service is a kind of command because it embodies an application behavior in one class, instead of a method. It can be considered also as a decorator because it allows adding new features (properties and behaviors) to an application in a non intrusive way. Services may be plain behaviors that are triggered as a consequence of some user action or might involve a navigational presence, i.e. a set of pages with information or operations corresponding to the service (as in Figure 1). We are particularly interested in this last kind of volatile services. Given a new (volatile) requirement we first model its conceptual and navigational features in a separate layer using the OOHDM approach. A second step is to indicate the relationships among services and existing conceptual and navigational classes; Figure 3 shows a preliminary specification of this connection. In the left we show the base model containing core (stable) application abstractions and in the right we present the specification of the service.

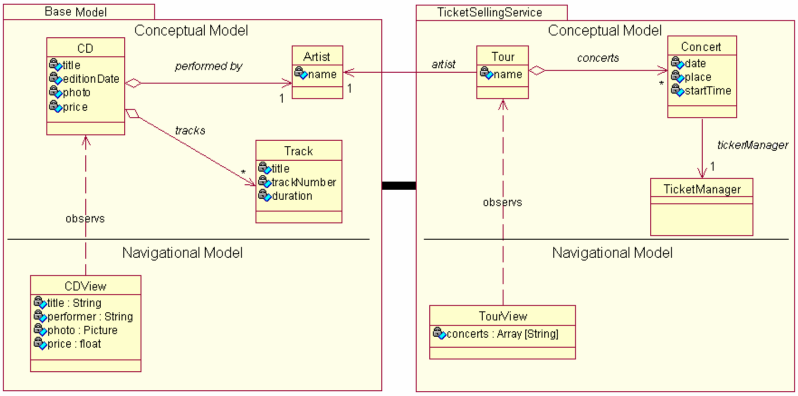


Fig. 3. Separating Volatile services from the base model

Notice the knowledge relationship among the Tour object and the performer's CD which inverts the "naive" solution in which Artists know the tour (thus coupling both classes), and the absence of link between the node CD and the Tour node. While the former is characteristic of Decorators, i.e. we are wrapping the model with a new service, the latter gives us the flexibility to specify different navigation strategies; for example we can either link the new functionality from the application (Figure 2) or insert it in the base node (Figure 1).

3.4 Integrating Volatile Services into the Core Design

VServices are connected to the application level using an integration specification, which is decoupled both from services and base classes. This specification indicates the nodes that will be enhanced with the volatile service, and the way in which the navigation model will be extended (e.g. adding a link, inserting new information in a node, etc). Notice that in the previous examples we aimed at extending only some specific instances of the CD (respectively Book) nodes. For example we might want to link some Rolling Stone's CD's to the ticket selling services for a concert in Argentina.

We define the affinity of a service as the set of nodes (respectively objects) in the design model which will be affected by the services, i.e. those nodes from which we will have access to the service. According to [8] we specify the affinity of a service in terms of objects' properties. Affinities are specified using a query language similar to the one that OOHDM itself uses for nodes specification [12] which was inspired in [4]. Those nodes which match the query are affected by the service. A query has the form: FROM *$C_1 \ldots C_i$* WHERE *predicate* in which the C_j indicate node classes and the predicate is defined in terms of properties of the model. Queries can be nested and a generic specifier (*) can be used to indicate that all nodes can be queried. As in OOHDM, the qualifier *subject* allows to refer to conceptual model objects. A query indicates the kind of integration between application nodes and services which can be *extension* or *linkage*. An *extension* indicates that the application node is "extended" to contain the service information (an operations) while, in a *linkage* the node "just" allow navigation to the service. For example:

Affinity Concert
From CDView *where* (performer = TourView.subject.artist.name)
Integration: Linkage (TourView.name)

The affinity named Concert (corresponding to the example in Figure 2) indicates that all instances of a CD node will have a link to those instances of TourView such that the performers are the same. The link is enriched with the name of the tour. Service might of course have more than one instance; for example in the case of the second motivating volatile functionality, many artists may be on tours. Each tour ticket selling functionality has its own data and the most important remark, may have its own integration style into core nodes. Thus, we may have to specify an affinity for each service instance, which is called an Instance Affinity to differentiate it from a Class Affinity. The functionality in Figure 2 has the following integration rule:

Instance Affinity Concert
Where (artist=U2 and TourView.subject.location= "Argentina")
Integration: Extension (TourView)

An affinity specifies a temporal relationship between a service and the model which can be evaluated either during the compilation of the model, thus requiring re-compilation each time the affinity changes, or can be evaluated dynamically during page generation, as will be explained in section 4. Notice that model objects (conceptual and navigational) are oblivious with respect to services and their affinities and then they can evolve independently of their volatile nature.

3.5 Further Issues

We treat services as first class objects in our model. In consequence we can define services which apply to a service also using affinities, and therefore composing services in a seamless way, without a need to couple services with each other. A nice example is the following: Suppose that we want to offer a travel service in our e-store; the service may be a general one, i.e. accessible as a landmark (See for example www.amazon.com) or it must be accessible only when certain offers arise. For example, we could offer those people who buy tickets to a concert our travel service when the service takes place in a particular city. In this case we will specify an affinity between the travel service and the ticket service as for example:

Affinity TravelService
From TourView *where* (Subject.concert.place= "Paris")
Integration: Linkage (TicketView)

Once again we obtain a clear separation between services and their target objects (being them base application nodes or services). The Travel service can be used in multiple other situations just by specifying corresponding affinities. A Service can be used for example in the context of a business process activity, e.g. as defined in [11], just by indicating the affinity and the target node (e.g. an activity node in the check-out process).We have also defined the concept of service specialization (a kind of inheritance in the realm of services) but for the sake of conciseness we omit to discuss this here.

4 Architectural Design and Implementation

We have implemented a framework on top of Apache Struts which supports semi-automatic translation of OOHDM models, including the instantiation of Web pages, from the OOHDM navigational schema and their integration with volatile services. The framework also provides a set of custom tags to simplify user's interface development according to the guidelines of OOHDM's abstract interface specification. A high-level description of the framework's architecture is depicted in Figure 4.

Our light-weight framework aims to:

- Allow the specification, and the straightforward implementation, of a web application navigational model, which contains nodes and links primitives such as those defined in OOHDM.
- Provide support for dynamic integration of volatile functionality.

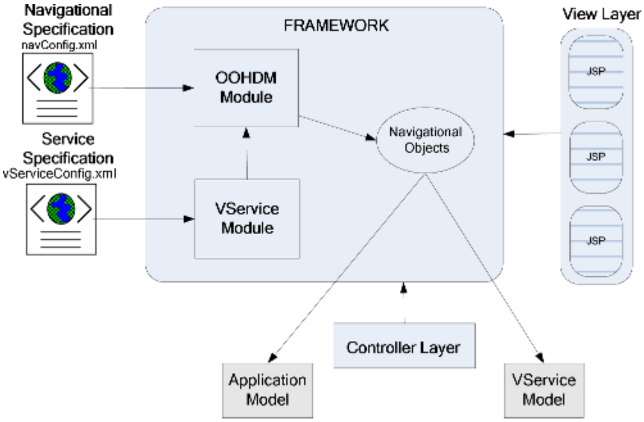


Fig. 4. Architecture of a framework for volatile services

The OOHDM module provides tools to represent a navigational layer between application domain objects and the user interface. We use the standard Struts controller objects to act as navigation controllers and to perform the interaction with conceptual objects. In this module the developer defines actions and links which allow representing the concepts in a navigation schema. This module offers support for defining nodes which contain the information which will be displayed in a page and profits from Struts custom tags for defining interface issues. Nodes contain Struts actions to manage navigation logic which is completely delegated to the Struts basic engine. The OOHDM module receives the navigational model in the form of a configuration file (NavConfig.xml) in which the designer specifies nodes, links and other navigation primitives. The information is processed and transformed into navigational objects which constitute the navigation layer of the application.

The volatile service module supports the integration of volatile functionality in a non-intrusive way; i.e. by releasing the developer from re-factoring existing classes or configuration files. This module is in charge of administrating and gluing volatile services in the target application, and uses the OOHDM module as a collaborator, delegating controller and navigation tasks to it. As mentioned before, a service is composed of a set of navigational nodes and conceptual objects that comply with a specific requirement. Nodes affinities are computing according to the actual state of the node's context which is defined as the set of direct and indirect relationships with other nodes and conceptual objects. The developer also provides all service information through a configuration file.

5 Related Work

Volatile requirements have been recently dealt by the requirements engineering community; particularly in [10] the authors propose to use aspectual constructs to isolate volatile requirements, mainly when they cross-cut "core" application behavior. Our approach also aims at separating these concerns, but without using aspects.

Web Engineering Methods have already faced the problems of e-commerce applications. Particularly, OOHDM [11] and UWE [5] have enriched their modeling armory for representing business processes. These methods have also defined means to personalize general application behavior and specifically business processes. OOWS [9] has also exemplified many of their features for specifying complex functionality using e-commerce software as a target. None of these methods have already explicitly dealt with volatile functionality. However, OOWS has been recently extended to incorporate external services in the context of business processes using a model-driven approach [13]. In [1], the authors present an aspect-oriented approach for dealing with adaptivity. In both cases, the concept of affinity could be easily introduced to mediate in the context of service integration in OOWS or adaptive aspects weaving in UWE.

6 Concluding Remarks and Further Work

In this paper we have presented an approach for dealing with volatile functionality in Web applications, i.e. for integrating those services which arise during evolution and are either known to be temporary or are being tested for acceptance. Incorporating this functionality in the conceptual and navigational model of a Web application might cause maintenance problems due to the need of editing classes which work properly or to clutter the existing model with possible spurious classes. We propose to add a separate layer for specifying volatile functionality. We have exemplified our approach with some simple examples and presented a way to integrate the VService Layer into the core application schemata, by using the concept of affinity. Affinities, which are expressed as queries, allow connecting services into those application objects which fulfill the desired properties. We have briefly described an implementation architecture that supports the evaluation of affinities and the injection of components defined in the VService layer into the core application objects.

We are studying the implication of service inheritance and composition and analyzing the integration of external services (e.g. Web Services). We are currently testing the described framework with demanding applications (e.g. those in which heavy queries must be executed). We are also studying the process of service integration via re-factoring of model classes.

References

1. H. Baumeister, A. Knapp, N. Koch and G. Zhang. Modelling Adaptivity with Aspects. 5th International Conference on Web Engineering (ICWE'05). Springer Verlag, Lecture Notes in Computer Science.
2. M. Douglas, D. Schwabe, G. Rossi, “A software arquitecture for structuring complex Web Applications” Journal of Web Engineering, Rinton Press, September 2002.
3. E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns. Elements of reusable object-oriented software, Addison Wesley 1995.
4. W. Kim, "Advanced Database systems", ACM Press, 1994.
5. N. Koch, A. Kraus, C. Cachero, S. Meliá: Modeling Web Business Processes with OO-H and UWE. 3rd International Workshop on Web Oriented Software Technology (IWWOST03), Oviedo, Spain, 2003.

6. Koch, N., Kraus, A., and Hennicker R.: The Authoring Process of UML-based Web Engineering Approach. In Proceedings of the 1st International Workshop on Web-Oriented Software Construction (IWWOST 02), Valencia, Spain (2001) 105-119
7. OMG Model-Driven-Architecture. In http://www.omg.org/mda/
8. M. Nanard, J. Nanard, P. King: IUHM: A Hypermedia-based Model for Integrating Open Services, Data and Metadata. Proceedings of Hypertext 2003; ACM Press, pp 128-137.
9. O. Pastor, S. Abrahão, J. Fons: An Object-Oriented Approach to Automate Web Applications Development. Proceedings of EC-Web 2001: 16-28
10. A Rashid, P Sawyer, AMD Moreira, J Araujo Early Aspects: A Model for Aspect-Oriented Requirements Engineering. Proceedings of RE, 2002, pp 199-202.
11. H. Schmid, G. Rossi: Modeling and Designing Processes in E-Commerce Applications. IEEE Internet Computing, January/February 2004.
12. D. Schwabe, G. Rossi: An object-oriented approach to web-based application design. Theory and Practice of Object Systems (TAPOS), Special Issue on the Internet, v. 4#4, October, 1998, 207-225.
13. V. Torres, V. Pelechano, M. Ruiz, P. Valderas: "A Model Driven Approach for the Integration of External Functionality in Web Applications" Proceedings of MDWE 2005. ICWE 2005 Workshop on Model-Based Web Engineering.
14. The UML home page: www.omg.org/uml/
15. A. Van Lamsweerde: Goal-Oriented Requirements Engineering: A Guided Tour Fifth IEEE International Symposium on Requirements Engineering (RE'01) p. 0249
16. D. Zowghi, A Logical Framework for Modeling and Reasoning About the Evolution of Requirements Proceedings of the 4th Pacific Rim International Conference on Artificial Intelligence, Cairns, Australia, 1996.

Design of Ubiquitous Referral Marketing: A Business Model and Method

Kyoung Jun Lee and Jong Chul Lee

School of Business, Kyung Hee University
Hoegi-Dong, Dongdaemun-Ku, Seoul, Korea
{klee, mermaio}@khu.ac.kr

Abstract. This paper provides a corporation's marketing strategy under a ubiquitous computing environment: a WOM(word-of-mouth) marketing using RFID(Radio Frequency Identification) technology and a business model which facilitates the word-of-mouth marketing. To this end, we examine the word-of-mouth communication effects on consumers' life, changes in corporations' attitude toward word-of-mouth marketing, and the difficulties that corporations have in conducting word-of-mouth marketing. The business model this paper suggests makes seamless business-to-consumer and consumer-to-consumer networking possible using the RFID technology and facilitates the word-of-mouth marketing through incentive system of each economic player.

1 Introduction

Word-of-mouth Marketing is marketing a good or a service by the message spread by customers where the communication takes place voluntarily and informally between people or groups. It can also be referred to Referral Marketing in that consumers refer to the experience of other consumers in the process they start to be aware of a product or a service, form an opinion, and finally make a purchase. In under a ubiquitous computing environment, the gap of media between the real world and information system can be narrower with the emergence of various computing terminals embedded in mobile network and sensors. All the information can be delivered seamlessly among economic players who are engaged in every commercial activity [3]. The existing communication methods used in online and offline word-of-mouth can evolve into a new form of communication that uses every channel of word-of-mouth, either online or offline, so any information can be sent and received seamlessly.

In under a ubiquitous computing environment, all goods have IDs in a form of RFID, Ipv6, or color codes etc. Since digital information is embedded in goods in the first place, information can be delivered seamlessly without any cost to convert analog information into digital one. In addition, it is easy to figure out the origin of information. Using such features, a company can use a product as a source to deliver product information, and a product plays a role of a medium to link products and consumers. In the end, a company is able to establish a system that controls marketing messages and reward customers for advertising products. WOM Marketing provides

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 102 – 111, 2006.

an experience of a prior purchaser based on trust between consumers. No matter how open a social boundary a consumer belongs to is, social network is formed on the basis of trust between people. The trust between consumers has a positive effect on the trust of an information receiver. But negative words of mouth spread faster than positive words and information can be distorted in the process of dissemination [1],[5]. In comparison, information spreads without distortion under a ubiquitous computing environment. In other words, there is little need for trust between consumers in this ubiquitous computing environment.

The WOM marketing under a ubiquitous computing environment creates less transaction costs, provides new efficacy to those who are involved, and make a transaction process transparent. But the use of auto-identification such as RFID could end up as infringement of privacy of the purchaser. This is a negative effect of using auto-identification. A customer can provide information about the clothes (s)he is wearing to many people and get rewarded, but what (s)he possesses and how much they are could be exposed to anyone. Therefore, a prior purchaser should have the right to decide whether the information will be provided to random potential consumers or not, and the identity of the prior purchasers should not be accessible to potential consumers who scan the information.

This paper suggests a new form of word-of-mouth communication emerging in the ubiquitous computing environment, along with a new business model and methods that facilitate the new communication.

2 WOM (Word-of-Mouth) Marketing

Words of mouth play a pivotal role in spreading a new product and have a defining effect on consumers' selection of products among various groups. Approximately 80% of the consumers are still affected by the recommendation of other people when they decide to buy something [8]. The purchasing pattern of today's consumers hinges not only upon the communication between a company and a consumer, but upon communication network of consumers based on user experience. In this section, we will explain the limit of current WOM communication.

Scenario 1

People around Jane always ask about where she bought the wardrobe she wears and how to coordinate clothes. Then Jane kindly gives tips about the brand, price, and the store information of the clothes she buys. Afterwards, people buy the same or similar clothes as Jane's at the same store Jane bought them. And they get information on clothes and how to coordinate clothes from the homepage of the brand on the Net.

This scenario depicts a situation that takes place in our daily lives where WOM communication is carried out. In this scenario, Jane plays a role of an "employee" who promotes the clothes of the company by wearing them. However, the store she bought her clothes does not pay her anything even though she played a role in increasing sales. In the same token, if Jane is dissatisfied with her clothes, people who have been talking to her could have had a negative image about the clothes and its brand [1]. In other words, people who had a positive image about the brand could turn out to have a negative one after talking to Jane. Therefore, companies need to come up with a system to reward customers like Jane who voluntarily promotes their products as well as measures to capitalize on those good customers. Such a system should encourage current good customers to recommend their products and make potential customers to accept the recommendation positively and purchase the products.

Scenario 2

Tom went skiing in an outworn ski suit and he thought he should buy a new one. Tom was lining up for a lift and he found Jay's ski suit and liked it. So he decided to buy the same ski suit with him and remembered the color and design. He could have asked Jay its brand and price, but he shied away from asking him those things because he was a total stranger. Tom went back to his room and searched on an online shopping mall. He failed to find a ski suit he wanted and gave up.

Anyone could have experienced the situation in this scenario. One day, you see something you like on the street, but you have no one to ask where (s)he bought it and don't know the brand and the place that sells the good you like. Sometimes you cannot find it even after you search for it on the Net. In this scenario, Tom could not find the ski suit on the Internet. The emergence of the Internet gave consumers greater access to product information, and consumers' search cost has dramatically been reduced, but on the other hand, search cost remains in one way or another since consumers have to find the information that suits their needs among the overwhelming volume of information. Moreover, Tom searched only ski suits that are registered online, and if Jay's ski suit is not registered online, it is impossible for Tom to find it. The impeded flow of information between online and offline is an obstacle to seamless business activities. In addition, the seller who sold Jay a ski suit and does not provide a channel that is easily accessible to consumers lost a potential customer (Tom). Therefore, both the seller and Tom need a new word-of-mouth channel that reconnect the severed flow of information between online and offline.

3 Ubiquitous Referral Marketing

In scenario 2, if Jay's ski suit had a RFID tag on it, and Tom possessed a mobile handset embedded with a RFID module, things could have been different. Tom could have scanned the RFID tag on Jay's ski suit with his mobile handset and gained the information he wanted and known where to go to find that suit, online or offline. In the meantime, the seller can secure a potential customer without any advertisement. In scenario 1, if people around Jane had gained information on Jane's clothes via the RFID tag hidden in clothes, they would get product information no matter what opinion Jane has about the clothes. Such a WOM communication is different from the one in the past in that information is disseminated via RFID tags, not people's mouth.

In a ubiquitous computing environment, all goods have IDs in a form of RFID, Ipv6, color codes, and since digital information is embedded in goods in the first place, information can be delivered seamlessly without any cost to convert analog information into digital one. In addition, it is easy to figure out the origin of information. Using such features, a company can use a product as a source to deliver product information, and a product plays a role of a medium to link products and consumers. In the end, a company is able to establish a system that controls marketing messages and reward customers` for advertising products.

WOM Marketing provides an experience of a prior purchaser based on trust between consumers. No matter how open a social boundary a consumer belongs to is, social network is formed on the basis of trust between people. The trust between consumers has a positive effect on the trust of an information receiver. In comparison, information spreads without distortion under a ubiquitous computing environment, the content of information can be disseminated as a company intended. In other words, there is little need for trust between consumers in this ubiquitous computing environment. A potential customer can actively receive based on what he sees and feels. The dissemination of information takes a form of "pulling" than "pushing." The "pulling" of information takes a similar form to benchmarking [4] which is a process in which a company compares its products and methods with those of experts, prior purchasers, or community members in order to try to improve its own performance. The only difference is that an information giver involves less in the formation and delivery of the information under a ubiquitous computing environment, and companies can remove variables that were out of control in the past word-of-mouth marketing.

The ubiquitous referral marketing under a ubiquitous computing environment creates less transaction costs, provides new efficacy to those who are involved, and make a transaction process transparent due to the features described above. But the use of RFID could end up as infringement of privacy of the purchaser. This is a negative effect of using RFID. A customer can provide information about the clothes (s)he is wearing to many people and get rewarded, but what (s)he possesses and how much they are could be exposed to anyone. Therefore, a prior purchaser has the right to decide whether the information will be provided to random potential consumers, and the identity of the prior purchasers should not be accessible to potential consumers who scan the information.

In chapter 4, we will propose a new economic player, a new business model and a method that enable companies to enhance the performance of their marketing activities and reduce the risk of privacy infringement using the characteristics of ubiquitous referral marketing.

4 Business Model

According to the business model definition by Timmers [7], the business model we propose is composed of four business entities as seen in below.

- ✓ Prior purchasers (Jane in scenario 1, Jay in scenario 2): The Consumers who purchase products online or offline. They register what they purchase by opening an account with ubiquitous referral marketing network (a new economic player this paper proposed) of a seller.
- ✓ Sellers (Shop in scenario 1): People who sell products embedded with a RFID tag whether they take an online or offline commerce form. They can secure potential customers who are interested in the products.

- ✓ Potential customers (people around Jane in scenario 1, Tom in scenario 2): They are consumers who have a mobile terminal with RFID Module and are interested in the products that prior purchasers possess. They can immediately check the information on the product they are interested in by scanning the RFID tag without any search cost. And they, using mobile network, can obtain additional information that sellers provide via RN.
- ✓ RN (ubiquitous referral marketing network): It plays a middleman who relays information between sellers and consumers. A seller provides additional information on the product in response to the request from a potential customer and gives some reward to a prior purchaser who possesses a product that the potential customer scanned information from.

Figure 1[1] shows the structure of the roles of a seller, a prior purchaser, a potential customer, and RN and the flow of information between them.

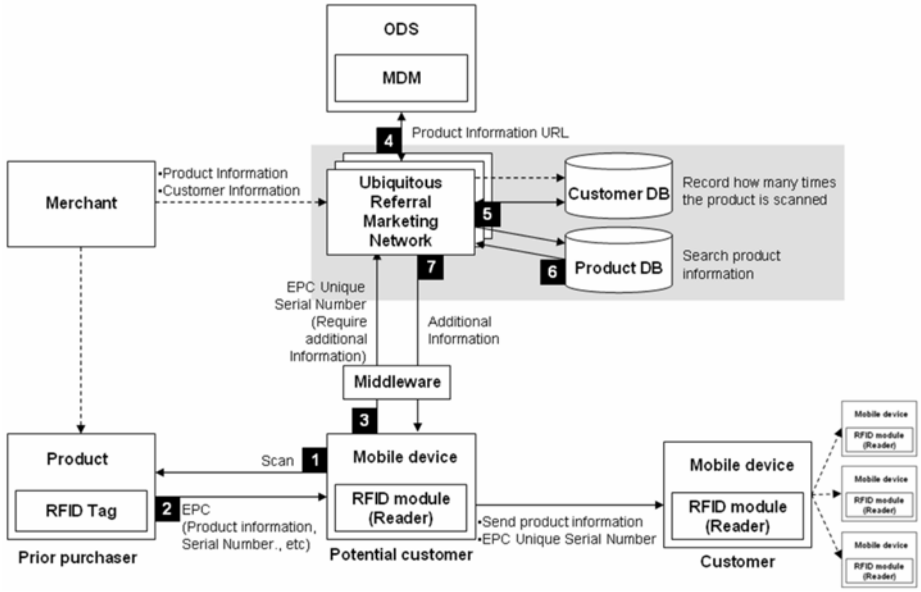


Fig. 1. Ubiquitous referral marketing service architecture

The following explains how information spread from a prior purchaser to an interested person or a potential customer and to another potential customer in turn.

A prior purchaser buys a product embedded with an RFID tag from a merchant and open an account with the advertising server the merchant is operating and registers the product (s)he bought on the server. The RFID Tag contains basic product information as well as EPC with which a potential customer can obtain additional information.

[1] This architecture is designed based on EPCglobal network and RFID ODS (Object Directory Service) of NIDA (National Internet Development Agency of Korea).

① A potential customer scans the RFID tag hidden in the product with a mobile RFID module-embedded handset.

② RFID tag scanned by the potential customer gives him or her basic product information.

③ If the potential customer needs more information, (s)he requests additional information with filtered EPC unique serial number to the RN through mobile networks.

④ The RN requests the product information URL to ODS[2] server. Decoding RFID tag through MDM[3], ODS server provides the location of server containing product information such as price, store location, and users' review related to RFID tag.

⑤ Set up standardized incentives granted to the prior purchaser who has registered a product depending on the number of times the EPC unique serial number is transmitted in response to an information request.

⑥ The RN searches for product information requested by the potential customer in a product database using the EPC unique serial number they referred to.

⑦ The RN transmits the findings to the potential customer who requested information and their mobile handset displays the received product information.

A potential customer is able to transmit the information to other consumers with a proper mobile handset and they can obtain additional information in the same way.

In this business model proposed above, a prior purchaser can be a potential customer of another seller at the same time. In other words, a consumer can play a role of both a prior purchaser and a potential customer. When an individual register as a seller on RN, a consumer can play three roles at once. RN, as a middleman between sellers and consumers, provides incentives to prior purchasers for advertising a specific good. In this way, the aversion to the RFID tag can be subsided. It is sellers who give out such incentives. Sellers can have easier access to consumers, especially potential customers who are interested in their products. Therefore, the cost is worth it.

This system has a similar cost/profit structure to CPC(Cost-Per-Click) search commercial model of Overture(http://www.overture.com) which offers money depending on the number of visitors to the commercial websites registered on the search engine. In addition, RN can provide a variety of services besides mediating between sellers and customers. Upon the permission of customers, it can detect their tastes and interest based on the collected data on their purchase and information requests. In other words, "pushing" type of advertising using customer profiles is possible as well as "pulling" types in response to information requests from customers. Besides, RN can offer numerous services including comparison shopping, price search, and display of relevant products. Accordingly, sellers can carry out various marketing activities targeting their potential customers along with word-of-mouth marketing, capitalizing on the RN.

[2] RFID ODS(Object Directory Service) provides the location of server containing product information related to RFID Tag using DNS Technology.

[3] MDM(Multi-code Decoding Module) is decoding module for promoting interoperability among several RFID codes such as EPC of VeriSign, ISO/IEC code and U-code of uID center.

The first element of the business model proposed in this paper is described above and the second and third ones can be explained as in Table 1.

Table 1. Sources of potential benefits and profits for each economic player

	Potential benefits	Source of revenues
Prior purchaser	- Stock management with RFID Tag. - Better services from sellers through RN.	-Incentive
Seller	-Reduction in marketing cost -Get more potential customers.	-Increasing profits
Potential customer	-Reducing search cost by networking with RFID.	-Incentive
Ubiquitous referral marketing network	-New business opportunities	-Registration fee -Cost-Per-Purchase revenue

5 Working Condition for the Business Model

In this section, we will find some conditions, for RN-registered sellers competing with RN-unregistered sellers. RN-registered sellers are those who register the ubiquitous referral marketing network (RN) and RN-unregistered sellers are the online sellers who do not have the ubiquitous referral marketing network membership.

Notations

- ✓ $Price_{reg}$ = The product unit price of RN-registered seller.
- ✓ $Price_{unreg}$ = The product unit price of an online RN-unregistered seller.
- ✓ $Cost_{reg}$ = The product unit cost of the RN-registered seller reflecting the prime cost, shop operating cost(including RFID system), delivery cost, etc.
- ✓ $Cost_{unreg}$ = The product unit cost of a online RN-unregistered seller reflecting prime cost, shop operating cost, delivery cost, etc.
- ✓ CAC_{reg} = A unit customer acquisition cost of the RN-registered seller including advertising cost and RN membership fee.
- ✓ CAC_{unreg} = A unit customer acquisition cost of an online RN-registered seller.
- ✓ SC_{reg} = Shopping cost incurred to the customer when (s)he purchases a product from the RN-registered seller through the RN including delivery cost, seller trust cost, and search cost etc.
- ✓ SC_{unreg} = Shopping cost incurred to the customer when (s)he purchases a product from a online RN-unregistered seller including delivery cost, seller trust cost, and search cost etc.
- ✓ RN_{Fee} = The cost that the RN-registered seller pays to the RN when a transaction occurs through the RN. There are a variety of methods to calculate RN_{Fee} such as CPC(Cost Per Click), CPM(Cost Per Mile) and CPA(Cost Per Action) in online advertising.
- ✓ INCEN = The money (incentive) that the RN-registered seller pays to a prior customer when a potential customer purchases a product by using RFID Tag through the RN. There are also a variety of methods to calculate INCEN.

5.1 The Condition for the RN-Registered Seller

In this section, we try to find the working condition for the ubiquitous referral marketing business model by comparing the RN-registered seller with a RN-unregistered seller who sells products through eBay Korea(www.auction.co.kr). The RN-registered seller should have profits through the ubiquitous referral marketing program (Condition 1). If the RN-registered seller and the RN-unregistered seller sell the same product at the same price, ubiquitous referral marketing is available when the total customer acquisition cost of RN-registered seller, i.e. including INCEN and RN_{Fee} and is less than that of RN-unregistered seller (Condition 2). If the both sellers sell the same product at different prices, ubiquitous referral marketing is available when the total cost is less than that of the online seller (Condition 3). Table 2 summarizes the costs and prices of the two kinds of sellers.

Table 2. Comparison between sellers

	RN-registered Seller	RN-unregistered Seller
Original Cost	$Cost_{reg}$	$Cost_{unreg}$
Cost after ad	$Cost_{reg}$ + CAC_{reg}	$Cost_{unreg}$ + CAC_{unreg}
Cost after sales	$Cost_{reg}$ + CAC_{reg} + INCEN + RN_{fee}	$Cost_{unreg}$ + CAC_{unreg}
Profit	$Price_{reg}$ - $Cost_{reg}$ - CAC_{reg} - INCEN - RN_{fee}	$Price_{unreg}$ - $Cost_{unreg}$ - CAC_{unreg}

$$\text{Profit} = Price_{reg} - Cost_{reg} - CAC_{reg} - INCEN - RN_{fee} > 0 \quad (1)$$

$$\text{if } Price_{reg} = Price_{unreg}, INCEN + RN_{fee} + CAC_{reg} < CAC_{unreg} \quad (2)$$

$$\text{if } Price_{reg} \neq Price_{unreg}, Cost_{reg} + INCEN + RN_{fee} + CAC_{reg} < Cost_{unreg} + CAC_{unreg} \quad (3)$$

5.2 The Condition for the Prior Customer to Join

To join this business model, customers have to get more benefit than the costs for offering personal information and joining ubiquitous referral marketing. There are two kinds of benefit that customers can get: potential benefit and practical benefit. The potential benefit is the monetary interest, that is, the incentive that customers can get by handing over RFID Tag information to others. This monetary interest can be divided into several kinds by the ways of giving incentive. For example, incentive can be given whenever a potential customer asks for additional information to RN using RFID Tag or by purchasing results. In addition, the practical interest comes from purchasing a product with RFID Tag. The customer can do the stock management of the product and get information service that is similar to the ones from online shopping malls. Besides, the potential benefit of the prior customer gets bigger by ‘network effect’, as the diffusion rate of mobile device with RFID Module gets higher and it becomes a cultural code to get information of product by scanning its RFID Tag so that more and more people use it. The incentive that the prior customer gets can greatly increase when the potential customer hands over the Tag information to others after using it to get information of the product.

5.3 The Condition for the Potential Customers to Join

The condition for the potential customers to join this business model is $\text{Price}_{\text{reg}}$ + SC_{reg} < $\text{Price}_{\text{unreg}}$ + SC_{unreg}. As the price of RFID Tag gets lower and the reader diffusion rate gets higher, the difference in search cost will decide whether a customer joins ubiquitous referral marketing or not. As we can see in the two scenarios mentioned above (Jane's friends and Tom used RFID Tag to solve the problem that could not be solved in traditional commerce environment), ubiquitous referral marketing with RFID technologies brings convenience and lower search cost at the same time.

6 Conclusion

The ubiquitous computing environment is expected to provide a contact point for companies to have access to more customers [2]. RFID technology has been introduced into the process of manufacturing and logistics very quickly, but it needs to anticipate the process after a purchase is made. The ubiquitous referral marketing delivers advertising messages via products people are carrying with them as described above and at the same time it can use a thing that doesn't belong to anyone. For example, consumers can get information about a book such as synopsis, a list of relevant books, and review of other readers by scanning the RFID tag of the book you are interested in. Street advertising banners or school boards can go beyond simply putting advertising message and extract immediate reactions from consumers by offering an opportunity to get additional information. The ubiquitous referral marketing will offer goods and services that fit consumers' needs by analyzing changes of emotions consumers feel in a situation as well as tastes and preferences.

This paper proposes a business model that is designed to use RFID technology from the perspective of marketing in the process after a purchase. However, there are many technical difficulties for every seller to carry out ubiquitous referral marketing independently, and the cost-efficiency is unpredictable. Moreover, consumer information is hard to get especially when they react very sensitively to the exposure of their privacy [6]. Therefore, a new business player will emerge to support sellers with technology and connect them with consumers through marketing activities, and in the meantime to protect consumers' privacy while rewarding them for their contribution to sales. Furthermore this business model can create value from both ways by helping consumers in searching, comparing, and selecting products based on their taste and assisting companies in making decisions on manufacturing and logistics. To make this business model a success, it is important to make consumers to perceive the whole process of gaining information by scanning RFID tags and incentives being granted to the information providers as one of the purchasing patterns and have no aversion to the system.

Acknowledgments

This research is supported by the Ubiquitous Autonomic Computing and Network Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

References

1. Chatterjee, Patrali, “Online Reviews – Do Consumers Use Them?" ACR 2001 Proceedings, eds. M. C. Gilly and J. Myers-Levy, Provo, UT: Association for Consumer Research, 129-134, 2001.
2. Fleisch, E., & Tellkamp, C., “The Challenge of Identifying Value-Creating Ubiquitous Computing Applications”, Workshop on Ubiquitous Commerce, UbiComp 2003.
3. Holtjona, G. & Fiona, F.. “U-Commerce: emerging trends and research issues,” Industrial Management & Data Systems, Vol. 104. No. 9, pp. 744-755, 2004.
4. Hisao, N, "Marketing strategy in the era of ubiquitous networks", NRI Papers, No.44 March 1, 2002.
5. Richins, M. & Marsha, L., “Negative word of mouth by dissatisfied consumers: A pilot study”, Journal of Marketing, 47(1), pp. 68-78, 1983.
6. Roussos, G. & Moussouri, T. “Consumer perceptions of privacy, security and trust in ubiquitous commerce”, Pervasive and Ubiquitous Computing, Vol. 8. pp. 416–429, 2004.
7. Timmers, P., “Business Model for Electronic Markets, Electronic Markets”, Vol. 8, No. 2, pp. 3-8, 1998.
8. Chung, J. & Kim, Y., “An Analysis of WOM Effects on the Consumer Product Choice by Using a hierarchical Bayesian Probit Model”, Korean Marketing Research, Vol. 19, No. 3, pp.1-20, 2003.

Pre-service and Post-transcoding Schema for an Adaptive PC to Mobile Web Contents Transcoding System*

Euisun Kang, Daehyuck Park, and Younghwan Lim

Department of Media, Soongsil University, Seoul, Korea
kanges86@naver.com
{hotdigi, yhlim}@ssu.ac.kr

Abstract. A factor to be considered in browsing of an existing web page to a mobile terminal is the difference in hardware environments between average PCs and terminals. It is necessary to service contents effectively in accordance with diverse environmental information of various terminals caused by the rapid development in communication devices. This various terminal information needs much time to generate the content for a server and much capacity to load in the server. Therefore the method to minimize response time and server capacity is required. This paper proposes a pre-service and post-transcoding method to provide a more rapid response time according to requirements of various terminals and illustrates the results of test system.

1 Introduction

Wireless Internet technology not only supplies a variety of services based on the Internet but also iscreases the maxmum of applications. It also supplies Internet service for mobile terminals (cell phones, PDA, and various other devices) using a wireless communication environment [1]. However, it is necessary to adapt content that is to be serviced by considering the characteristics of client terminals in order to supply an original web site for PCs. These mobile terminals, such as cell phones, PDAs, and other devices have heterogeneous environments. There are various methods to adapt contents. A re-authoring method is to reconstruct contents serviced in a PC web browser for mobile device contents [2][3]. Another method is to transcode contents into the language that can be recognized in a mobile device [4][5]. Others are the way to transcode and provide multimedia resources existed in a PC web for mobile device in real-time [6][7], to provide service which exhibits a fast response by considering the mobility of a terminal [8] and so on..

The most important factor to be considered in the service of the web contents for a mobile terminal is the variety of terminals, such as markup language, resolution, color depth, memory size, etc. The number of mobile terminals rapidly increased and many kinds of terminals came out and changed according to the development of communication devices. Thus, it is necessary to provide effective services for these various

[*] This work was supported by the Soongsil University Research Fund.

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 112 – 121, 2006.

terminals. That's because of too much time to generate contents in real-time and a variety of mobile terminals. In addition, the amount of mobile contents that is transcoded in each terminal occupies some capacity in a server according to the number of mobile pages. Therefore, a method which minimizes response time and server capacity according to transcoding time is required.

This paper proposes an adaptation system to service usual PC web pages to mobile terminals and analyzes the response time in this system. And this paper proposes a pre-service and post-transcoding method to provide faster response time for a mobile terminal to solve the problem produced from the analyzed data.

2 An Adaptive PC to Mobile Web Contents Transcoding System

2.1 System Architecture

In this paper, an adaptation system is implemented using the concept introduced in [9][10][11][12][13] to adapt wired web pages to various mobile terminals. The adaptation framework designed in this paper is named a Mobile Gate System. Fig. 1 represents the structure of this system.

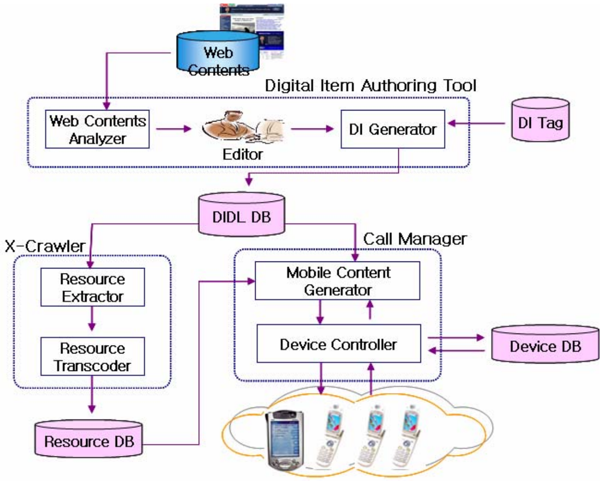


Fig. 1. Mobile Gate System Architecture

As a type of editor, Digital Item Authoring Tool assists in reconstructing various web contents for mobile use by selecting the only resource that will need to be displayed on the PDA or mobile phone from web content displayed on the PC. This component is done in off-line. The produced mobile web contents are automatically created in the type of DI which satisfies MPEG-21 standard. In this tool, Web Contents Analyzer parses PC Web pages wrapped by HTML or XML, and then divides the web page into resource and expressive specification and assists users to choose the resource more conveniently. Using the extracted and arbitrary resource, Editor helps the user construct the mobile web contents more simply through a copy and paste. DI Generator creates the reconstructed mobile page into MPEG-21 DI format using DI Tag Table and saves it in DIDL DB. X-Crawler is a part which carries out transformation in advance regarding actual resource in DI edited by administrator. This component is performed in Off-line. Resource Extractor parses DIDL generated by authoring tool and extracts information relative to the resource from DIDL. Among resources which need transformation, Resource Transcoder performs the actual transcoding by using the transformation information. It is a part of the Resource Adaptation Engine of DIA (Digital Item Adaptation) in MPEG-21. Resource DB saves and manages the information of transcoded resources. In case that a mobile terminal requested server, Call Manger module finds out the characteristic of device in Device DB, reconstructs web documents which browser device is able to recognize and then sends them. By applying the multimedia data which has been converted according to each device platform, Mobile Contents Generator creates documents suitable for each device in XSL. It is similar to the Description Adaptation Engine of DIA in MPEG-21 DIA.. Device Controller analyzes information about mobile device which is requesting service and manages the Device DB.

2.2 Problem

Response time is one of the factors to be considered to service multimedia data efficiently for various mobile terminals. Thus, the MobileGate system proposed in this study is applied to check the response time of a server when certain web pages are required to service in a mobile terminal. One is the way of the transcoding of all mobile contents in an off-line state and the other is the way of trascoding content in an on-line state using a call manager in real time. Fig. 2 represents the results of this investigation.

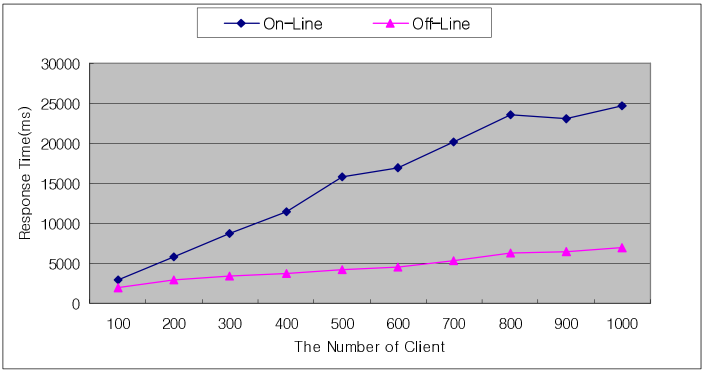


Fig. 2. Response time for Generating Mobile Content

As shown in Fig. 2, the direct service using a pre-transcoding process for the content applied in each terminal demonstrates a fast response time because it does not need transcoding and generating contents. However, because the server analyzes a terminal's information and generates appropriate mobile contents in real-time, it is evident that the response time is increased continuously in proportion to the number of terminals in an on-line state even though the multimedia data (some images are used in this case) is pre-transcoded in an off-line state. Moreover, the capacity of the server is continuously increased.

In order to solve these problems, this paper proposes a transcoding method that first provides available service required in a mobile terminal regardless of the quality of service. In other words, after pre-service is provided, the mobile content is transcoded for the terminal later.This way is used to apply caches and reduce the server capacity and response time.

3 Pre-service and Post-transcoding Method for Access Technology of a Mobile Terminal Using PC Web Contents

A cache replacement algorithm [8][14][15] and prefetching method [16][17] are used to minimize the response time on a PC web page. However, it is difficult to apply these ways to a mobile service because the variety of mobile terminals is not considered. The simplest way to provide high quality service to a terminal is to generate appropriate content considering the analysis of the QoS (Quality of Service) of a terminal. However, this method has a problem. Too much cost is required to generate contents including descriptions (i.e., markup language) and metadata of resources in real-time. Thus, this chapter proposes a pre-service and post-transcoding method based on the level of service satisfaction.

3.1 Pre-service and Post-transcoding Processes

A pre-service process contains a content cache storing already generated content to provide faster service and service proper content for a terminal by searching it in the cache when service is required from a terminal. The LFU(Least Frequency Used) is used as a content cache replacement algorithm. If there is no content which corresponds to a terminal in the cache, the optimal content but with lower qulity resource can be serviced is selected and serviced in advance instead of exactly matching content. Therefore, we are named Pre-Service.

The post-transcoding process is a type of transcoding method that transcodes content after pre-service to a terminal in preparation for re-request of the same content when the required service is not provided, or the most similar content is serviced within the playable range.

Fig. 3 illustrates the process of the pre-service and post-transcoding process. The Pre-Threshold is the range that can be serviced in a pre-service process when the required content from a terminal is searched in a content cache. In order words, Pre-Threshold defined a lower oundary on acceptatble QoS. Post-Transcoding Threshold is a high-qulaity boundary that a post-transcoding can be required within the Post-Threshold. The QoS (Quality of Service) is a factor that determines playability, such as markup, resolution, color depth, and supported image formats. Pre-Threshold and Post-Trhshold are inputted by system user.

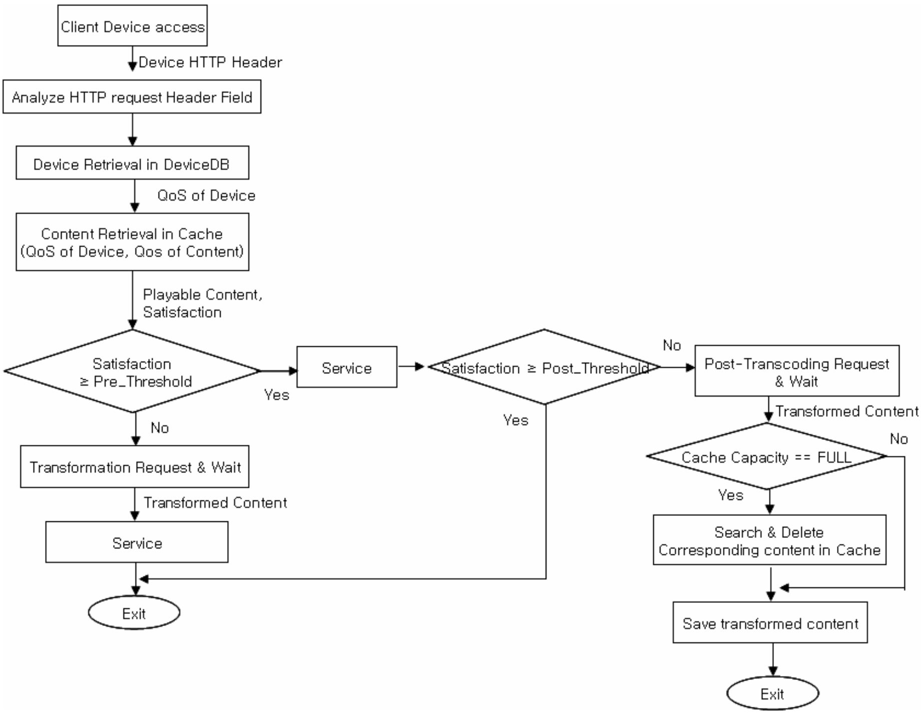


Fig. 3. The Process of Pre-Service and Post-Transcoding

Algorithm : Pre-service and Post-Transcoding

```
Step 0. Initialize the Pre-Threshold and Post-Threshold as System Parameters.
Step 1. Call Manager receives certain service requirements from a specific terminal.
Step 2. Call Manager analyzes the HTTP reques theader field in the terminal.
Step 3. Search DeviceDB by using the analyzed HTTP Header.
Step 4. Search a corresponding content to the QoS information that is returned
        from the searched DeviceDB in the cache. Calculates the level of service
        satisfaction by comparing the QoS of the terminal with the searched content in the cache.
Step 5. Check whether the calculated degree of service satisfaction is larger than
        the Pre-Threshold or not.
        IF satisfaction ≥ Pre _Threshold
                Go to Step 6.
        Else
                ① Require a transcoding process to generate proper content that is
                   appropriate to terminal because the searched mobile content
                   cannot be played on that terminal.
                ② Wait for completion of the transcoding process and terminate
                   the process after providing service.
```

```
Step 6. Service currently searched content, then exits.
Step 7. Search whether the level of service satisfaction is larger than the
          Post_Threshold or not.
      IF satisfaction $\geq$ Post_Threshold,
            Terminate the process and wait for next request of a terminal.
      Else
            Requires a post-transcoding process for the content within the range
            of the Post -Threshold.
Step 8. Check the cache in order to store the transcoded content in the cache.
      IF Cache capacity == FULL
            Searches the content stored in the cache by using a replacement al-
            gorithm and deletes it.
Step 9. Store the result and terminate the Call Manager and then wait for following
          instructions.
```

The Pre-Service and post-Transcoding method is used to reduce loads caused by in real-time content transcoding. It is possible to reduce the response time by performing pre-service according to the degree of service satisfaction even if optimal service is difficult when an appropriate content does not exist in the cache of the presently requested content.

3.2 Measuring Method of the Degree of Service Satisfaction

As illustrated in Fig. 3, when mobile device request content, a server searches for an appropriate content in a content cache. In this process, if the appropriate content does not exist in the content cache, a server searches a playable content and services it in advance. Because mobile content is a description generated by a transcoding process using DIDL based on the QoS information(resolution, color depth, sound poly and ,etc) in a device, the essencial elements for reproduction can be extracted and stored. The degree of service satisfaction of a terminal, which presently requires service and mobile contents in cache can be expressed as a probability. The premise in the measurement of the degree of service satisfaction is presented as follows:
mc_i is the first mobile content stored in content caches.

$$\textit{Premise condition: QoS of } mc_i \leq \textit{QoS of device}$$

This is because the content, which has larger QoS information than that of the terminal that requires a service, cannot be serviced to the terminal. For instance, an image with a resolution of 320*240 cannot be serviced to a terminal with a resolution of 176*144.

Simple Method

A simple method to assure a level of service satisfaction uses a probability by mapping the QoS information extracted from the mobile content and that of a terminal at the rate of 1:1 as follows:

$$Satisfaction(mc_i, md) = \left(\sum_{k=1}^{k=QoS} \frac{QoS_k \ of \ mc_i}{QoS_k \ of \ \ md} \right) / \ QoS_n$$

md is a client terminal, QoS_n is a number of QoS information of device, and QoS_k is the k^{th} information in the n QoS. If the content stored in a cache is serviced according to the calculated degree of service satisfaction, service can be provided within the degree of service satisfaction.

The Method Applied Weight

Mobile content is generated by the QoS information of a terminal. Priority can be given to this QoS. For instance, a call manager generates markup language provided first for efficient service if the markup language provided to the terminals is different. And it is necessary to transcode and service multimedia data according to the resolution of terminals. A weight applied method is how to grade importance on provided QoS information. In this way, the degree of service satisfaction is calculated effectively. The following expression is the way to calculate the degree of service satisfaction.

$$Satisfaction(mci, md) = \left(\sum_{k=1}^{k=QoS} \frac{QoS_k \ of \ mc_i \times \omega_k}{QoS_k \ of \ \ md \ \times \omega_k} \right) / \ QoS_n$$

ω_k is the weight that is given according to the importance of the QoS_k information. A service method using the degree of service satisfaction has some advantages.It copes with the content that doesn't exist in a cache flexibly by considering the limited content cache. And the efficiency of a server is improved by minimizing the transcoding time in the description.

4 Results of Experiment

The system proposed in this study was tested by using a Windows 2000 server that has a 1.8 GHz Pentium IV CPU and 512 MB memory. A digital item authoring tool was used to evaluate the efficiency as illustrated in Fig. 1, and a DIDL as an intermediary file was generated. And some mobile contents were produced by using the Call Manager and X-Crawler.

Fig. 4 illustrates response time according to the probability range of the degree of service satisfaction and measuring method of the degree of service Satisfaction. Here, 80%+Wegiht indicates that service satisfication is calculated by using weight and the mobile content which satisfied 80% of the multiple contents existed in the caches is supposed to service. Of course this is applied when there is no content which corresponds to the terminal in the cache.

From the above picture, we can see that it requests more response time when using Weight applied method than when using simple method. The reason is that complexity for calculating weight and accuracy of satisfaction by grading weight is added in Weight applied method compared with Simple method. And the response time for probability range of service satisfaction in different measurement methods is the least in case of 70%. The reason is that there are more contents of 70% satisfaction than contents of 90% satisfaction in Cache.

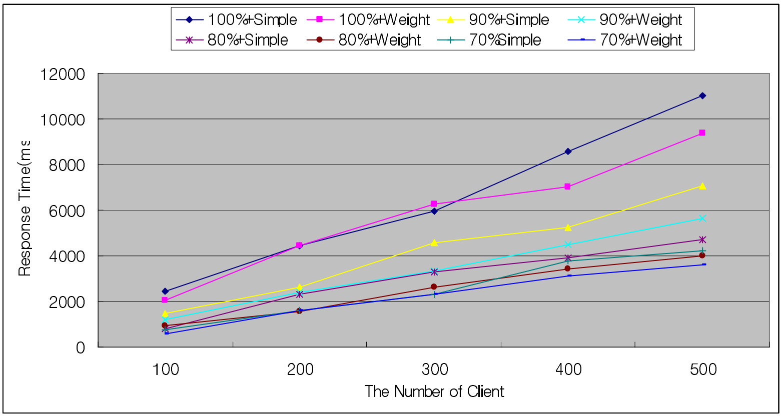


Fig. 4. The response time by service satisfaction and the range of probability

Fig. 5 illustrates the comparison of the response time between the proposed pre-service and post-transcoding and the On-line service. In here, On-line service is to transcode content using a call manager in real time without cache and Pre-serivce & Post-transcoding is to be suggested in this paper. To consider accuracy and response time, Weight applied method and 80% probability range is used.

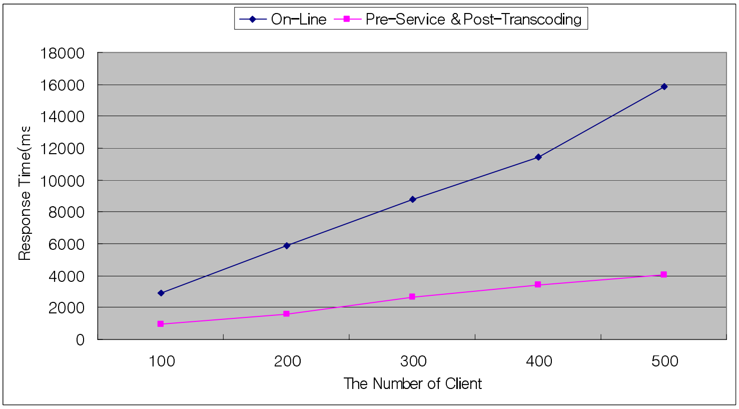


Fig. 5. The response time of on-line serivce and pre-service &post-transcoding

As described in Fig. 5, Pre-Service and post-Transcoding has 60% faster response time than on-line service method. It may be caused by the effect of cache, but providing the content can be serviced in advance can reduce response time. Although it can't service the fittest content for connected device, it can reduce latency time. In addition we can generate the fittest content for the same device before. So we can offer effective service when the same device is connected later.

5 Conclusion

In a ubiquitous environment wireless Internet has been trying to provide users with necessary services efficiently whenever and wherever. However, there are some difficulties in the production of contents suitable for users' requirements because of the variety of mobile communication environments. In addition, it is hard to avoid the costs needed to develop and maintain such contents. In order to solve these problems, this paper proposes a mobile gate system, which supplies web pages for PC to mobile devices such as cell phones and PDAs. A pre-service and post-transcoding method based on playable service satisfaction for a mobile terminal is proposed to solve the problems. From the performance estimation, we can confirm that Pre-Service and post-Transcoding can improve response time compared with existing methods. Pre-Service and post-Transcoding compares and analyzes information of devices and contents, and then first services within the playable range. Therefore it can improve latency time. Also, this system can prepare for the same device which has been connected before, so it can cope with rapid changing devices' information.

References

1. Goodman, D.J. The Wireless Internet: Promises and Challenges. IEEE Computer, pp. 36-41, Volume 33, No. 7, July 2000.
2. Xing Xie, Gengxin Miao1*, Ruihua Song, Ji-Rong Wen, Wei-Ying Ma. : Efficient Browsing of Web Search Results on Mobile Devices Based on Block Importance Model. 3rd IEEE Int'l Conf. on Pervasive Computing and Communications(PerCom 2005), Kauai Island, Hawaii, March 2005. pp17-26
3. Younghyun Hwang, Jihong Kim, Eunkyong Seo. : Structure-Aware Web Transcoding for Mobile Device. IEEE, Seoul National University, Seoul, Korea, September 2003, pp14-21.
4. E. Kirda, C. Kerer and G. Matzka. : Using XML/XSL to build adaptable database interfaces for Web site content management. XML in Software Engineering Workshop, 23rd International Conference on Software Engineering, Toronto, Canada, 2001.
5. WebSpere:http://www-3.ibm.com/software/info1/websphere/index.jsp
6. Maria Hong, DaeHyuck Park, YoungHwan Lim. : Transcoding Pattern Generation for Adaptation of Digital items Containing Multiple Media Streams inUbiquitous Environment., ICCSA International Conference, Singapore, LNCS 3583, pp.1036-1045, 2005.
7. Euisun Kang, Maria Hong, Younghwan Lim.: A Guided Search method for Real time Transcoding a MPEG2 P frame into H.263 P Frame in a Compressed Domain. ICCSA International Conference, Singapore, LNCS 3581, pp 242-251, 2005.
8. Ariel Pashtan, Andrea Heusser. : Personal Service Areas for Mobile Web Applications. IEEE Internet Computing, November, 2004. pp34-39
9. Chua H. N.a, Scott S.D.b, Choi Y. W. c, and Blanchfield P. : Web-Page Adaptation Framework for PC & Mobile Device Collaboration, 19th International Conference on Advanced Information Networking and Applications(AINA'05), 2005, pp727-732

10. Chan Young Kim and Jae Kyu Lee, Yoon Ho Cho, Deok Hwan Kim. : VISCORS: A Visual-Content Recommender for the Mobile Web. IEEE Intelligent System, December 2004.
11. Yong-hyun Whang, Changwoo Jung, et al. : WebAlchemist: A Web Transcoding System for Mobile WebAccess in Handheld Devices. SPIE Vol. 4534, p.37-47, 2001.
12. I. Burnett et al. : MPEG-21: Goals and Achievements. IEEE MultiMedia, vol. 10, no. 6, Oct-Dec. 2003, pp. 60-70.
13. MPEG MDS Group. : Information technology - Multimedia framework (MPEG-21) - Part 2: Digital Item Declaration. ISO/IEC TR 21000-1:2005, Final Draft.
14. D. Lee, J. choi, J. Kim, S. Noh S. Min Y. Cho, and C. Kim. : LRFU replacemen policy:a spectrum of block replacement policies. IEEE Transactions on Computers, vol. 50, no. 12, pp 1352 - 1361, Dec. 2001.
15. Kai Cheng and Yahiko Kambayashi. : Advanced Replacement Policies for WWW Caching. 1st International Conference on Web Age Information management(WAIM), pp 21-23, Shanghai, China, June 2000.
16. Wei-Guang Teng, Cheng-Yue Chang, Ming-Syan Chen. : Integrating Web Caching and Web Prefetching in Client-Side Proxies. IEEE Transactions on parallel and distributed system, vol. 16, No. 5, May 2005.
17. Q. Yang and HH Zhang. : Integrating Web Prefetching and. Caching Using Prediction Models. World Wide Web, vol. 4, no. 4,. pp. 299-321, 2001

Context-Aware Recommendation Service Using Multi-leveled Information in Mobile Commerce*

Joonhee Kwon[1] and Sungrim Kim[2]

[1] Department of Computer Science, Kyonggi University,
San 94-6, Yiui-dong, Yeongtong-ku, Suwon-si, Kyonggi-do, Korea
kwonjh@kyonggi.ac.kr
[2] Department of Internet Information, Seoil College,
49-3, Myonmok-dong, Jungrang-Ku, Seoul, Korea
srkim@seoil.ac.kr

Abstract. Recommender systems are being used by an ever-increasing number of electronic commerce applications to help consumers find products to purchase. Recommender systems in mobile commerce should be context-aware to understand each consumer's contexts anywhere at anytime. This paper proposes a new context-aware recommendation service that enables consumers to obtain relevant information efficiently by using multi-leveled information. This paper describes the recommendation method and presents application scenario that utilizes the method. Several experiments are performed and the results verify that the proposed method's recommendation performance is better than other existing methods.

1 Introduction

Nowadays, more and more services are available in the form of human-computer interactions, especially due to the increasing interest in mobile computing environment. The movement toward mobile commerce has allowed companies to provide consumers with more options. However, in expanding to this new level of customization, business increases the amount of information that customers must process before they are able to select which items meet their need. One solution to this information overload problem is the use of a recommender system [1, 2]. A recommender system can be defined as a system which has the effect of guiding the user in a personalized way to interesting or useful objects in a large space of possible options [3].

To be able to perform a personalized recommender system needs to understand the user's interest or preference. Human perception is greatly aided by the ability to probe the environment through various sensors along with the use of the situated context. In return, the context has a large influence on the interest and intent of one particular user. This causes the interest and intent of a user to vary dynamically over time. Context awareness is thus a major factor when dealing with recommendation services.

* This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (R04-2004-000-10056-0).

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 122 – 131, 2006.

One of the possible context definition states that it is “any information that can be used to characterize the situation of an entity, where an entity can be person, place, physical or computational object”[4].

The main considerations in context-aware recommendation methods are how accurately the method recommends information required by the users and how rapidly it recommends this information, even with large amounts of information to recommend [5]. In this paper, we suggest a new context-aware recommendation service using multi-leveled information for more rapid and accurate recommendations. We adopted multi-leveled information based on the observation that all of the information based on each user’s contexts does not need to be recommended at the one time. By using multi-leveled information, we obtain less irrelevant information progressively as we become nearer to the context of interest. This enables more rapid and accurate recommendations because all of information is not given at one time, thereby eliminating the problem of including irrelevant information in the context values at one time.

Our discussion will proceed as follows. Section 2 will give an overview of related works. Section 3 will discuss the recommendation method and algorithms. Section 4 will describe a mobile commerce scenario. Section 5 will discuss the experiments. Finally, section 6 will conclude the paper.

2 Related Work

The context-aware recommendation is an attempt to deliver personalized information that is most relevant to the user within the particular context of that moment in time. The most approaches simply match user profiles to context values [6]. Because they use only explicit user profiles, there are limitations to recommend useful information. Moreover, these methods are not concerned about recommending information rapidly.

An advanced approach is a context-aware cache based on a context-aware diary [5]. Based on the diary’s contents, the context-aware cache tries to capture the information that the user is more likely to need in future contexts. The cache makes more immediate recommendation and reduces the cost of recommending information by processing most of the work automatically. The context-aware diary stores past and future data that are explicitly informed. This approach, however, simply matches data from the context-aware diary to the current context when the context-aware cache tries to capture future data.

Another approach is proposed in [7]. To recommend information rapidly, it locally stores the information that the user is likely to need in the near future based on behavior patterns. To retrieve the information to be recommended to the user, the method uses data mining. Moreover, by using a multi-agent architecture, it allows continuous rapid and accurate recommendations, even with a change in the user’s context and solves problems with the limitations of user’s mobile device storage. However, this method retrieves all of the available information at one time causing unnecessary and slow recommendations. Furthermore, when a behavior pattern is not found, this method does not recommend information rapidly and accurately.

3 Context-Aware Recommendation Using Multi-leveled Information in Mobile Commerce

A context-aware recommendation using multi-leveled information is comprised of two algorithms. Some recent studies have considered the use of association rule mining in recommender systems [8]. In the first algorithm, the recommendation rules are extracted from shopping records using the association rule mining, where the left hand side of the rule is the context values.

```
Algorithm 1.
Begin
   Input   shopping record, support, confidence
   Output  recommendation rules
   Method  association rule mining
End.
```

Algorithm 1. Extracting recommendation rules

In the second algorithm, the recommendation information in the near future is prefetched from product catalogs using the current context values, level values and recommendation rules. As the consumer's context changes, new information may need to be recommended immediately. To achieve this, we locally store the recommendation information that the consumer is likely to need in the near future using the context window. In this paper, we call the window that includes context values with possibility of being used in the future as the “context window”. If a behavior pattern is found, the context values in the context window are determined by the behavior pattern. Otherwise, the values are context values within a certain difference of value from the current context values.

In context-aware computing, a mobile device is usually used but it has limited storage. This makes it quite difficult to locally store the recommendation information extracted in the second step. To overcome this difficulty, only the recommendation information that will be used in the very near future is stored and gradually updated as the context changes.

The level value means the degree of relevance. The relevance is the confidence in the association rule mining [9], where the confidence is the degree of certainty of the association between context and recommendation information. The recommendation information needed for a low level value has a higher confidence and is broader than that needed in a high level value.

The level value determines the size of the context window in the very near future. The size of the context window becomes larger in inverse proportion to the level value. The level value of the current context value is determined by a levelizing policy. An example of a levelizing policy states that the level value may be determined using the time it takes to walk around in a certain location. That is, a short (or long) time to walk around in a certain zone is considered as a request for recommendation information with a confidence of over 80% (or 50%) on two (or one) zones so that the level value is 1 (or 2). Algorithm 2 shows the prefetching method.

```
Algorithm 2.
Begin
  Input    context value, level value, recommendation rules
  Output   recommendation
  Method
    While (contexts in the near future is discovered)
    Begin
      If ( level value by previous context value is greater than
           or equal to level value by current context value )
        If ( context values to be added by change of the level
             value exist )
            1. Extract context values in the future to be added.
            2. Extract action parts in the recommendation rules
               that the values of condition part are equals to
               the context values to be added.
            3. Extract only recommendation information greater
               than or equal to minimum confidence allowed in
               current level value in previous step 2.
        End If
      Else
         1. Extract action parts in the recommendation rules that
            the values of condition parts are equals to the
            context values prefetched already
         2. Extract only recommendation information greater than
            or equal to minimum confidence allowed in current
            level value, where extract only recommendation
            information lower than confidence allowed by the
            previous context values.
      End If
    End
End.
```

Algorithm 2. Prefetching

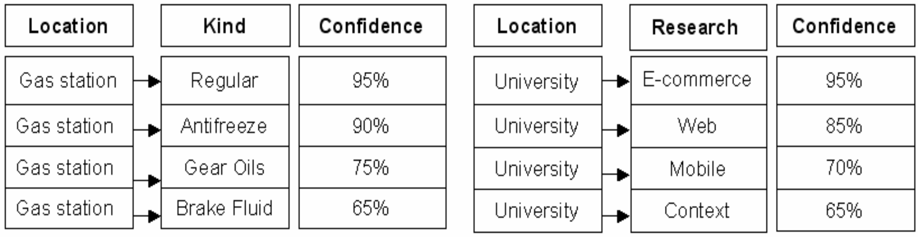
When a user's mobile device storage does not have enough space to store new recommendation results from Algorithm 2, the proposed method uses a replacement method. The higher the confidence of the information is, the higher the possibility that it will be accessed in the near future. We select the recommendation information in the lower confidence as the information to be replaced.

4 Mobile Commerce Scenario

In this section we present location-aware mobile commerce scenarios that utilize the proposed method. In this scenario, we compare the method in [5], which we call the existing method, and the proposed method. The scenario is described as follows:

A driver called Frank has a driving routine where he moves to the "Kyonggi University" from the "Shell gas station". He regularly fills up the car with gas in the gas station and does research in the university, as shown in Figure 1. Figure 1 and Figure 2 show the recommendation rules and the recommendation information from the first algorithm in Section 3 with a confidence of over 50%. He drives with a personal digital assistant (PDA) attached with the proposed recommendation method.

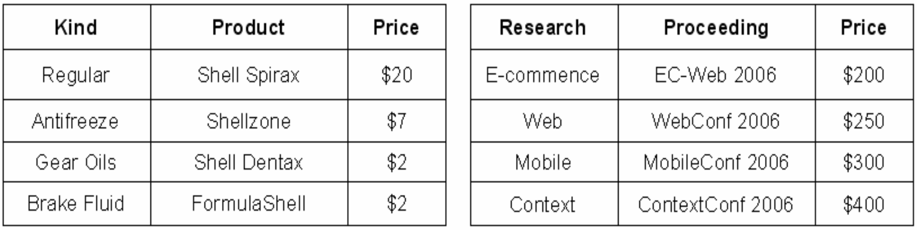
The system logs him in, responds with a welcome message, and then proceeds to present recommended driver information based on his interest and the location.



Location	Kind	Confidence
Gas station	Regular	95%
Gas station	Antifreeze	90%
Gas station	Gear Oils	75%
Gas station	Brake Fluid	65%

Location	Research	Confidence
University	E-commerce	95%
University	Web	85%
University	Mobile	70%
University	Context	65%

Fig. 1. Recommendation rules in application scenario

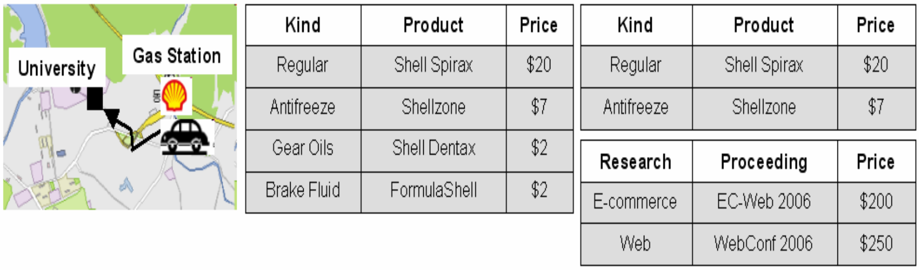


Kind	Product	Price
Regular	Shell Spirax	$20
Antifreeze	Shellzone	$7
Gear Oils	Shell Dentax	$2
Brake Fluid	FormulaShell	$2

Research	Proceeding	Price
E-commence	EC-Web 2006	$200
Web	WebConf 2006	$250
Mobile	MobileConf 2006	$300
Context	ContextConf 2006	$400

Fig. 2. Recommendation information in application scenario

The following are some assumptions. First, there are two level values. The level value is determined by the example of the levelizing policy in Section 3. Second, we set the maximum number of rows allowed in the Frank's PDA storage to 4.


University
Gas Station

(a) Around gas station

Kind	Product	Price
Regular	Shell Spirax	$20
Antifreeze	Shellzone	$7
Gear Oils	Shell Dentax	$2
Brake Fluid	FormulaShell	$2

(b) Existing method

Kind	Product	Price
Regular	Shell Spirax	$20
Antifreeze	Shellzone	$7

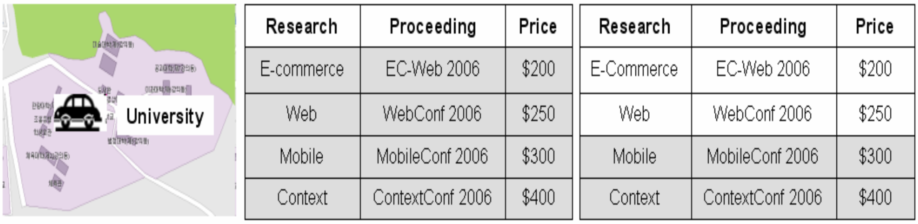
Research	Proceeding	Price
E-commerce	EC-Web 2006	$200
Web	WebConf 2006	$250

(c) Proposed method

Fig. 3. Recommendation information: "Around gas station" (Level 1)

Suppose that Frank's current location is around "Shell gas station" in Figure 3(a). Then, he drives to the "Kyonggi University" from the "Shell gas station", with fast velocity. By his PDA storage limitation, the existing method only extracts the information about gas station in Figure 3(b). Compared with Figure 3(b), the amount of information in Figure 3(c) related to gas station and university with a confidence of over 80% does not exceed his PDA storage capacity. This shows the result that the proposed method extracts and presents only the information with high confidence about the gas station and university. By driving fast, he does not receive any information with the low confidence about gas station, as shown in Figure 3. However, Figure 3(b) shows all of the recommendation information related to gas station, including information with the low confidence.

Finally, he moves to the "Kyonggi University", as in Figure 4(a). Suppose that his level value changed to two by driving slow in the "Kyonggi University". In Figure 4(b), he gets all of the information related to university at one time. Compared with Figure 4(b), the information related to university is gradually increased in Figure 4(c). The darkened parts in Figure 4(b) and 4(c) show the newly extracted recommendation information. We can observe that the amount of newly extracted recommendation information in Figure 4(b) is larger than in Figure 4(c). Resulting from this observation, the advantage of the proposed method can be seen: it retrieves a smaller amount of new information than the existing method.


University

Research	Proceeding	Price
E-commerce	EC-Web 2006	$200
Web	WebConf 2006	$250
Mobile	MobileConf 2006	$300
Context	ContextConf 2006	$400

Research	Proceeding	Price
E-Commerce	EC-Web 2006	$200
Web	WebConf 2006	$250
Mobile	MobileConf 2006	$300
Context	ContextConf 2006	$400

(a) University (b) Existing method (c) Proposed method

Fig. 4. Recommendation information: "University" (Level 2)

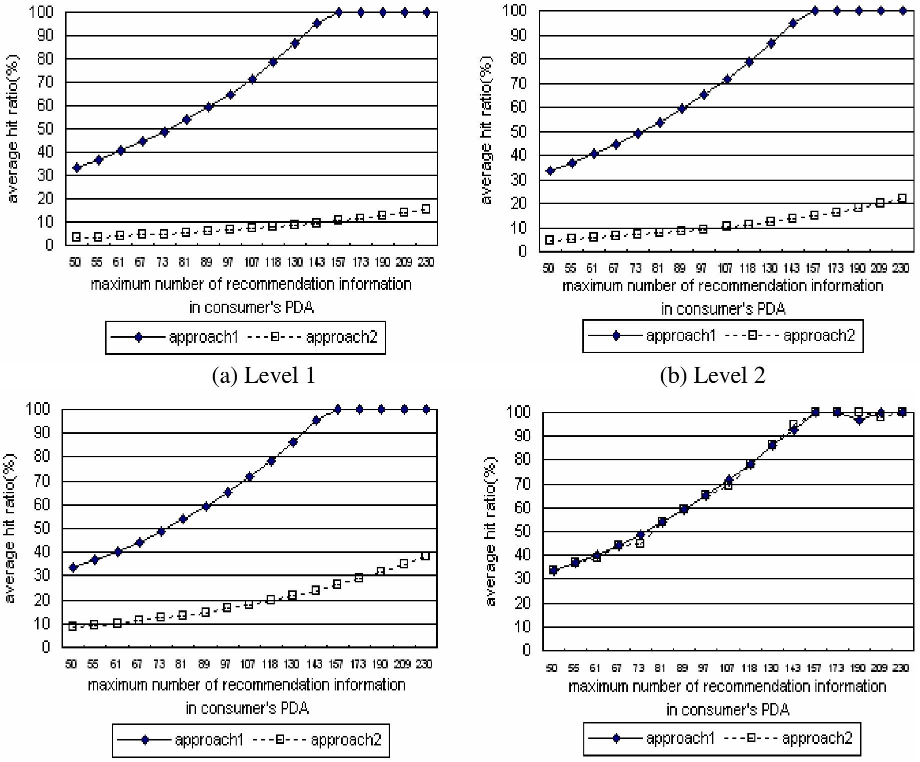
As shown in Frank's location-aware mobile commerce application scenario, the advantage of the proposed method is that it retrieves a smaller amount of new information than the existing method but with greater relevance.

5 Experiments

We implemented both the proposed method, called *approach1* and the existing method, called *approach2*. All programs were written in Visual C# .Net. The server ran on a Pentium IV desktop computer and the client ran on a Pentium laptop computer.

The contexts were generated from value 1 to the value 100 and the number of recommendation rules was set to the number of context values multiplied by the number of recommendation information. The number of recommendation information was set to the number of product catalogs in store multiplied by 0.005. The recommendation rules were randomly generated from the values of contexts and product catalogs, while the confidences of the rules were given randomly from 20 to 100. The number of levels in the generated data was 4. For each level, the size of the context window required in level value 1(2, 3, 4) was 10(7, 4, 1, respectively) and the confidence allowed in level value 1(2, 3, 4) was 80(60, 40, 20, respectively). In addition, the context values were not changed and level values were higher in our experiment. We ran the experiment 100 times for context values extracted at random number.

We evaluated the impact of the consumer's PDA capacity. We tested the average hit ratio and the average number of recommendation information read to compare rapid recommendation. The product catalogs in store were generated from value 1 to value 50,000. For comparison we varied the maximum number of recommendation information allowed in the consumer's PDA by 10% from 50 to 230.



(a) Level 1

(b) Level 2

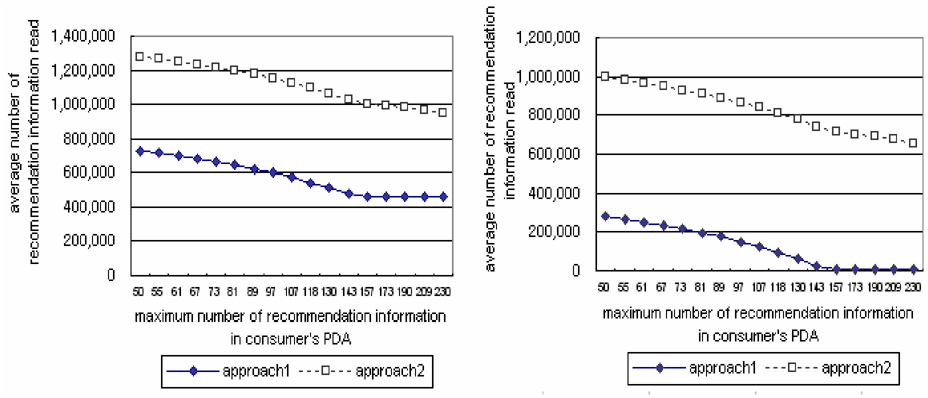
(c) Level 3

(d) Level 4

Fig. 5. Average hit ratio for each level

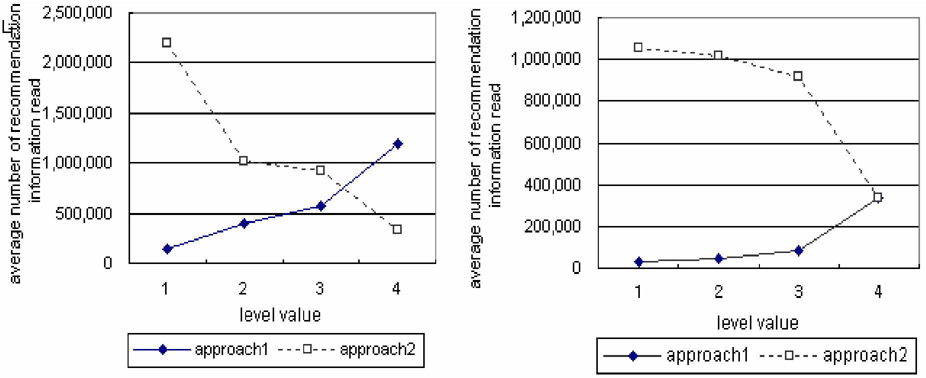
We measured the average hit ratio in the consumer's PDA for each context value to recommend information. As shown in Figure 5, for *approach1*, the average hit ratio is near 100% when the consumer's PDA capacity increases. However, average hit ratio for *approach2* does not exceed 40% in level value 1, 2, and 3. Only in level value 4, the average hit ratio for *approach1* and *approach2* is near same. In level value 4, both *approach1* and *approach2* use all the information; therefore, the results are almost the same. Several other observations are found in these results. First, the average hit ratio in *approach1* consistently performs better than that in *approach2*. Second, as level value lowers, the difference between *approach1* and *approach2* increases.

Figure 6 shows the average number of recommendation information read for each context value to retrieve the information whenever consumer's PDA capacity is increased. Figure 6(a) presents the result with prefetching and Figure 6(b) describes the retrieval performance after prefetching. Significantly, in Figure 6(a), unlike Figure 6(b), the performance difference between *approach1* and *approach2* is lower. This is due to the fact that approach2 extracts the most detail information at a time in level value 1 and then retrieves information much more from server by high miss ratio in consumer's PDA.



(a) with prefetching (b) after prefetching

Fig. 6. Average number of recommendation information read for consumer's PDA capacity



(a) with prefetching (b) after prefetching

Fig. 7. Average number of recommendation information read for each level

Figure 7 shows the average number of rows read in order to retrieve the information whenever the level value is increased. Figure 7(a) presents the result with prefetching, and Figure 7(b) describes the result after prefetching. In Figure 7(a), we can observe that the performance of *approach1* is better than that of *approach2* in level value 1, 2, and 3. Figure 7(b) also shows the performance of *approach1* is better than or equal to that of *approach2* in all levels. However, in level value 4 in Figure 7(a), the number of rows read in *approach1* is higher than that of *approach2*. *Approach1* prefetches the recommendation information with a lower confidence than confidence at the previous context value whenever the context values in the client's device storage are not changed and the level value is higher. However, *approach2* prefetches all the information at one time in level value 1. This makes the number of rows read in *approach1* higher than that in *approach2* in level value 4 in Figure 7(a).

6 Conclusion

Recommender systems address a variety of mobile commerce needs. Clearly one of the key requirements for mobile commerce is that the consumer experience be highly personalized. In addition, there can be a significant number of contexts with mobile commerce. The mobile commerce applications that account for relevant context characteristics will benefit from increased functionality and usability.

There have been some studies in context-aware recommendation methods. All of these methods, however, use all of the information available, including irrelevant information. We suggested a new context-aware recommendation service based on each consumer's context using multi-leveled information. We obtained information with lower confidence progressively as we got nearer to the context of interest, by using multi-leveled information. This enabled rapid and accurate recommendations because all of the information in the context value was not accessed at one time.

The proposed method had a number of advantages compared to existing methods. First, we presented a new context-aware recommendation service using multi-leveled information. Information was recommended rapidly and accurately, by using the multi-leveled information. Second, we illustrated location-aware mobile-commerce scenario that utilize the proposed method. Third, we showed that the proposed method's recommendation performance is better than other existing methods.

References

[1] J. Ben Shafer, J. A. Konstan and J. Riedl, "E-Commerce Recommendation Applications", Data Mining and Knowledge Discovery, Vol. 5, No. 1-2, p.115-153, 2001.

[2] P. Tarasewich, R. C. Nickerson and M. Warkentin, "Issues in Mobile E-Commerce", Communications of the Association for Information Systems, Vol. 8, p.41-64, 2002.

[3] N. M. Sadeh, Ting-Chak Chan, Linh Van, OhByung Kwon and K. Takizawa. "Creating an Open Agent Environment for Context-aware M-Commerce", Lecture Notes in Artificial Intelligence, p.152-158, 2003.

[4] A. K. Dey, "Understanding and Using Context", Personal and Ubiquitous Computing Journal, Vol. 5, No. 1, p.4-7, 2001.

[5] P. J. Brown, G. J. F. Jones, "Context-aware Retrieval: Exploring a New Environment for Information Retrieval and Information Filtering", Personal and Ubiquitous Computing, 2001, Vol. 5, Issue 4, p.253-263, 2001.

[6] G. Kappel, B. Proll, W. Retschitzegger and W. Schwinger, "Customisation for Ubiquitous Web Applications - A Comparison of Approaches", International Journal of Web Engineering and Technology, Vol.1, No.1, p.79-111, 2003.

[7] Joonhee Kwon, Sungrim Kim and Yongik Yoon, "Just-In-Time Recommendation using Multi-Agents for Context-Awareness in Ubiquitous Computing Environment", Lecture Notes in Computer Science 2973, p.656-669, 2004.

[8] B. Mobasher, R. Cooley and J. Srivastava, “Automatic personalization based on Web usage mining”, In Communications of the ACM, Vol. 43, No. 8, p.142-151, 2000.

[9] R. Agrawal, T. Imielinski and A. Swami, "Mining association rules in large databases", In Proceedings of ACM SIGMOD Conference on Management of Data, p.207-216, 1993.

Attribute-Based Authentication and Authorisation Infrastructures for E-Commerce Providers

Christian Schläger, Manuel Sojer, Björn Muschall, and Günther Pernul

University of Regensburg, Universitätsstrasse 31, D-93053 Regensburg, Germany
{christian.schlaeger, bjoern.muschall,
guenther.pernul}@wiwi.uni-regensburg.de

Abstract. Authentication and authorisation has been a basic and necessary service for internet transactions. With the evolution of e-commerce, traditional mechanisms for data security and access control are becoming outdated. Several new standards have emerged which allow dynamic access control based on exchanging user attributes. Unfortunately, while providing highly secure and flexible access mechanisms is a very demanding task, it cannot be considered a core competency for most e-commerce corporations. Therefore, a need to outsource or at least share such services with other entities arises. Authentication and Authorisation Infrastructures (AAIs) can provide such integrated federations of security services. They could, in particular, provide attribute-based access control (ABAC) mechanisms and mediate customers' demand for privacy and vendors' needs for information. We propose an AAI reference model that includes ABAC functionality based on the XACML standard and lessons learned from various existing AAIs. AAIs analysed are AKENTI, CARDEA, CAS, GridShib, Liberty ID-FF, Microsoft .NET Passport, PAPI, PERMIS, Shibboleth and VOMS.

1 Introduction and Motivation

E-commerce has become ubiquitous in today's world. One user maintains business relations with many vendors and owns numerous accounts and identities, each containing user profile data. The internet serves as a platform for all these business transactions. Securing these transactions is crucial for e-commerce providers [7]. In this environment, both, changing customer profiles and changing portfolios are putting pressure on functionalities.

Increasing demands on the customer side include:

- Growing heterogeneity and an increased number of user groups, comprising all socio-economic classes and diverging media literacy around the globe;
- Dynamic customer relations with changing customer data, identities, places of access, and credentials;
- Usability, privacy, and data security develop into buying criteria.

Increasing demands for resources include:

- Greater heterogeneity of offered resources (online services, brick-and-mortar products, Grid computing, communication services, etc.);

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 132 – 141, 2006.

- Greater complexity as a result of the distributed value chain in virtual organisations and embedded third-party suppliers;
- Changing portfolio of goods and personalised services.

On a technical level the demands mentioned here call for more flexible techniques for administering and managing resources and customers. Traditional authorisation models like RBAC or DAC cannot cope with such diversity while dynamic models like ABAC were designed for situation like these. ABAC can handle heterogeneous user groups and resources. In combination with an AAI Single Sign-On, resilient attribute exchange for authorisation and access control, the whole chain of security services can be outsourced. So far, however, no AAI/ABAC e-commerce application has been introduced yet.

ABAC is one of the latest developments in the field of authorisation and access control. With another concept, XACML (eXtensible Access Control Markup Language) [12], an open standard has been proposed by OASIS (Organization for the Advancement of Structured Information Standards) that is able to build complex policies that derive access control decisions from object and subject attributes. This standard is especially helpful in providing dynamic, flexible, and fine-grained access control for heterogeneous and vast user groups. The work of [19], [14], and [2] presented first applications using XACML.

AAIs provide several basic security services. Having developed from basic Single Sign-On solutions, they are able to manage authorisation processes and enforcement tasks today. Regarding AAI architectures, there are centralized approaches as well as federated concepts like the Liberty Identity Federation Framework. Comparative surveys on existing AAIs can be found in [15] and [9].

Using AAIs means sharing security information about subjects and objects with other service providers or central services. These attributes can be used in an access control model to define user privileges. Fig. 1 shows the process of granting access to resources with the help of user and resource attributes.

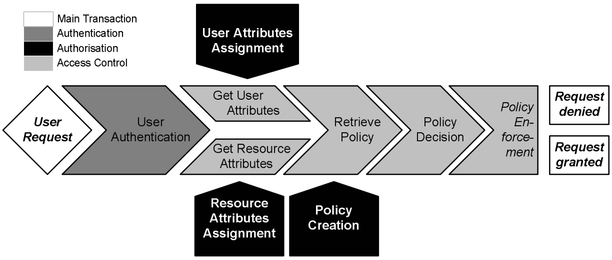


Fig. 1. Chain of Security Services in an Attribute Infrastructure

This work brings together open standards and existing AAI frameworks and products to merge functionalities and technical possibilities into a new AAI model including attribute based access control.

2 Requirements

Fig. 1 shows the generic chain of security services processing an access request. It is possible to outsource one single step from within the chain or any combination of two or more steps to an AAI. The more steps are outsourced the more powerful the infrastructure has to be. Fig. 2 shows a model to evaluate AAIs. The tasks outsourced to the AAI are listed cumulatively from the bottom up in this model, i.e. the higher an AAI is ranked the more tasks it will execute. The names of the steps are derived from SAML (Security Assertion Markup Language) [11] and the XACML standard.

In the easiest case the infrastructure provides only the Single Sign-On. At the next level the AAI covers the transfer of attributes about users and resources. This could be implemented as interfaces that allow a resource to query a user's home domain about attributes using SAML. Even more powerful are AAIs that can come to a decision regarding a user's access request and then forward this decision to the resource. Finally, at the fourth and highest level, it is possible for the AAI to enforce the decision by itself.

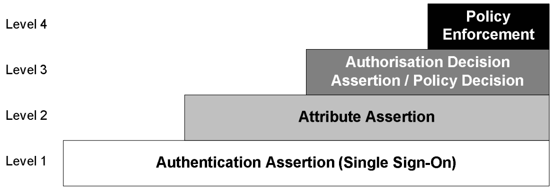

Level 4
Policy Enforcement
Level 3
Authorisation Decision Assertion / Policy Decision
Level 2
Attribute Assertion
Level 1
Authentication Assertion (Single Sign-On)

Fig. 2. AAI Levels

3 Clustering AAIs

AAIs can be grouped into three disjunctive clusters: WWW Systems, PMIs, and Grid Systems. AAIs in the first group are Liberty ID-FF, Microsoft .NET Passport, PAPI and Shibboleth. The PMI group consists of AKENTI and PERMIS while CARDEA, CAS, VOMS, and GridShib make up the last group. For the sake of completeness, Kerberos and SESAME have to be mentioned as AAIs, too. However, as they have either expanded into more recent systems (Kerberos) or are hardly in use anymore (SESAME), they will not be analysed in detail.

WWW Systems aim at providing a Single Sign-On point for several web applications [10] [5] [4]. In this scenario users benefit from the simplification of having to remember only one username and password for several applications. A similar goal is pursued by PMIs which, however, focus primarily on authorisation. They allow resources to outsource the formulation of an access policy and the access control decision process [17] [6]. The AAIs evaluated in the Grid cluster are add-ons to the already existing security architectures in Grids. Grid AAIs are highly sophisticated in handling authentication, but provide only coarse-grained and inflexible authorisation via so-called "gridmap-files". These files map Grid users to local users on the application system. Especially with regard to virtual organisations, this approach is suboptimal in terms of administration, scalability, and security [13] [1].

3.1 WWW Systems

There are two subgroups within this cluster. As members of the first group, Microsoft .NET Passport and the Liberty Alliance concepts deal mainly with authentication while the second group with PAPI and Shibboleth also offer authorisation services.

Microsoft introduced Microsoft .NET Passport in 1999 to offer a Single Sign-On service on the internet. When trying to log in at a resource the user is forwarded to the .NET Passport log-in page. The resource's Passport ID is transported to passport.com using URL encoding. If it is registered and valid the user is forwarded to passport.net. The user authenticates with his username and password and is redirected to passport.com. Passport.com writes four cookies in the user's browser cache. Following this, the user is forwarded to the resource. Finally, two more cookies are written to allow the user's Single Sign-On when he accesses the resource the next time [10].

In contrast to Microsoft .NET Passport, the Liberty Alliance develops only concepts and standards which allow compatibility between different implementations by third party companies. In general, Liberty offers the same functionality as Microsoft .NET Passport. However, it relies heavily on open standards like SAML and allows the use of federations and circles of trust between different authentication services [3].

As a member of the second subgroup, PAPI (Point of Access to Providers of Information) was developed by RedIRIS, a Spanish research network, in 2001 as a WWW System that also offers authorisation services. As a distributed access control system for information accessed via intra- or internet it is mainly used in libraries and digital archives. Following the PAPI processes, a user authenticates at the Authentication Server (AS) in his home domain. As PAPI is authentication agnostic, it is in the responsibility of the separate domains to authenticate their users with the appropriate means. Following successful authentication, the user is presented a website listing all digital resources he may access. Selecting one of these resources he is redirected to the Point of Access (PoA) guarding the respective resource. The link the user has clicked on contains an asymmetric key identifying the AS which has authenticated the user. Furthermore, it is possible to also add tokens to the link which transport attributes. If the key is valid, the PoA trusts the AS and thus also the user. The PoA evaluates the request according to the client's attributes, and queries the requested resource via an HTTP request. In doing so, the PoA acts as web proxy between the user and the resource. The retrieved resource is then sent to the user in combination with a new set of keys which allow the user to direct further queries to the PoA [5].

Quite similar to PAPI, Shibboleth was developed with the goal to protect digital resources from unauthorised access. Even though it provides less functionality, it is used more frequently, especially at American universities. In contrast to PAPI, Shibboleth does not offer a central starting point, but instead the user directly accesses a remote resource which then redirects him to their respective home domain for authentication. Following authentication, the user receives an opaque handle which is presented in front of the resource and allows the resource to query the user's home domain for attributes anonymously. These attributes then form the basis for the resource's access control decision [4].

3.2 Privilege Management Infrastructures

PERMIS and AKENTI [17] are members of the PMI cluster. They use X.509 attribute certificates (ACs) to store user attributes. In the following we will focus on the PERMIS processes.

PERMIS was developed as part of the ISIS project of the European Union and is currently being maintained by the University of Kent in the UK. A distinguished name (DN) allows access to a respective user's AC from a LDAP directory. This certificate lists the user's roles. The mapping between roles and privileges is described in a policy file formulated in XML. When trying to access a protected resource, the Access Control Enforcement Function (AEF) forwards the request of an authenticated user to the Access Control Decision Function (ADF). The ADF validates the user's certificate, interprets the policy, and forwards its decision to the AEF. The AEF enforces the decision and forwards the request to the resource if appropriate. Following the PERMIS concept, the AEF has to be implemented individually for each resource and is not included in PERMIS [6].

3.3 Grid Systems

CAS (Community Authorisation Service) has been under development since 2002 as an authorisation service for the Globus Toolkit and is part of its fourth version (GT4). CAS initially aimed at providing a superior authorisation solution to the coarse-grained and badly scaling on-board authorisation of the Globus Grid. When a user tries to access a CAS resource, he has to authenticate at the Globus Security Infrastructure (GSI) at first and can then request capabilities for the requested resource from the CAS server. If the requested capabilities are deposited at the CAS server, the server generates a signed assertion containing these capabilities. This assertion is linked to the user's Grid account. In order to finally access the resource the user authenticates again at the GSI and presents the access request together with the capabilities. If the capabilities suffice the resource will grant access [13].

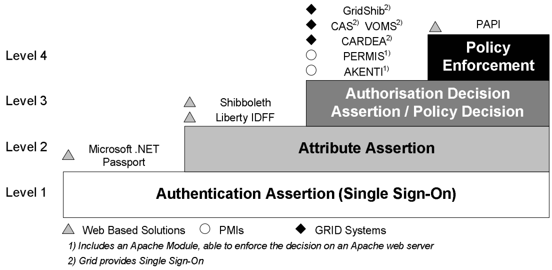
Quite similar in its architecture to CAS is Virtual Organization Management Service (VOMS). Yet, CAS allows more fine-grained authorisations than VOMS which relies on certifying group or role memberships [1].

Another resembling approach is presented in CARDEA which NASA developed for its Information Power Grid. CARDEA uses mainly open standards like SAML and XACML. Furthermore, CARDEA does not certify authorisation, but provides a "Yes" or "No" answer for access requests [8].

Finally, GridShib, which was started as a project in late 2004, is an attempt to transport the attribute architecture of the very successful Shibboleth to the Grid environment. Its general architecture is similar to that of CAS or VOMS, yet it allows authorisation based on attributes versus capabilities or roles and will provide anonymous authorisation services as does Shibboleth [18].

4 Matching AAIs and Requirements

Using the criteria from section 3 the analysed AAIs can be assigned to different levels according to Fig. 2. The result is presented in Fig. 3.



1) Includes an Apache Module, able to enforce the decision on an Apache web server
2) Grid provides Single Sign-On

Fig. 3. Clustering AAIs by capabilities

4.1 Authentication Assertion (Single Sign-On)

AAIs classified as SSO-Systems provide only authentication services. Adjacent services are left to the resource. The main idea is to enable the user to log-in into different systems with one username and password. However, each service provider keeps track of the user's account himself. Only the sign-on process is outsourced.

4.2 Attribute Assertion

Systems linked to the level "Attribute Assertion" provide Single Sign-On and are able to transport and assert user and resource attributes. The process of authorising access and enforcing access control decisions is again left to the resource, though. However, the AAI provides necessary information for this decision. Shibboleth and the Liberty ID-FF [3] use a protocol to exchange SAML attribute assertions with the user's home domain [4].

4.3 Authorisation Decision Assertion / Policy Decision

In addition to the aforementioned services, these AAIs are able to decide on the user's access request. The service provider retrieves the policy decision stating that the access is granted or not. The service provider is left with the enforcement of this decision. All analysed PMI systems and all Grid solutions fall into this category. However, the Grid solutions depend on authentication via the over-all Grid security architecture (e.g. GSI) and do not perform this task themselves [13] [1].

4.4 Policy Enforcement

Finally, on the highest level the AAI is not only responsible for authentication, attribute gathering, and policy decision but also for its enforcement. The complete chain of security services is handled by these systems. The only AAI in this section is PAPI. PAPI realises these services by acting as a proxy between user and resource. Consequently, all requests are intercepted by the proxy. If access is granted, the proxy forwards the request to the target system and returns the requested resource to the client [5].

5 Building a Generic ABAC Enabled AAI for E-Commerce

PAPI provides all identified security services as a proxy solution. However, due to its specialised architecture for online libraries it cannot be considered a generic solution for all different purposes, ranging from e-commerce to Grid computing and to inter-institutional collaboration. As a consequence, a generic model can use lessons learned from presented AAIs especially from PAPI, which seems to be a good starting point. To achieve a generic design, the heterogeneity of resources and consequently a complex attribute infrastructure has to be assumed additionally. Forms of federation for scalability like a Liberty circle of trust have to be integrated where reasonable. Another important requirement is to make use of open standards like SAML or XACML [5], when possible. Using the introduced systems and technologies as a basis, a best practice example can be assigned to all steps in the security service chain. Using these examples, a generic architecture for an attribute-based infrastructure is constructed in the following.

5.1 Authentication Assertion (Single Sign-On)

The process of a SSO is implemented most flexibly in the Liberty ID-FF. Every service provider can be chosen as a user's identity provider guaranteeing proper authentication [3]. However, if legal issues regarding security and liability in the case of misuse are important, a reliable provider is preferable. This service could be handled by various Identity Providers (IdP) being also Service Providers or specialised IdPs. We recommend using a Shibboleth-like "where-are-you-from"-server to locate a user's corresponding IdP [4]. Taking into account Microsoft's .Net Passport failure a variety of IdPs is preferable. They should only learn attributes about the user, not about the resource accessed or the computed access decision. This is in accordance with the separation of identity and attributes argued for in [7] and [15].

5.2 Attribute Assertion

Information about users and their behaviour in business transactions is confidential and must not be shared without filtering. Some of it is also constantly changing. Therefore, attributes about customers and resources have to be managed directly by the service provider. For privacy purposes there must be restricted release of attributes in form of an attribute release policy. With the recommended, user chosen IdP and a separated decision point we mediate between the user and the provider. In line with the XACML standard [12] we use Policy Information Points (PIP) to fetch attributes.

5.3 Authorisation Decision Assertion / Policy Decision

Within a federation one central policy covering access requests is not sufficient. However, in a distributed environment a central, high-level policy can be used for general requests and decisions. Such a policy must be enhanced by local, low-level and fine-grained policies tailored to specific needs and requirements. The XACML standard proposes a Policy Administration Point (PAP) to manage the consolidation of multi-layer policies. The Policy Decision Point (PDP) computes an access control decision based on a restricted set of attributes and policies. Additional environment information and metadata can be included in this process [14] [2]. Note that the PDP is able to compute a decision only on attributes, not knowing the identity of the user or the actual resource requested. We have used this advantage in our model.

5.4 Policy Enforcement

Using a proxy for the enforcement like in PAPI can be suitable. However, this decision is dependent on the use case and the security level. By nature, enforcing security decisions outside the target application neglects inherent information about the application. For fine-grained access control the application's context must be included. Therefore, target systems not including or requiring security functionality should outsource the enforcement to the infrastructure. If integrated security functionalities are needed or desired, the enforcement must be taken care of in the target itself. For a web based system like PAPI the usage of a proxy is reasonable.

5.5 Reference Model

Fig. 4 depicts the resulting reference architecture using SAML and XACML terms and definitions. The user directs his request directly to the service provider (SP). Following the idea of generic architecture for e-commerce environments, the user interacts with the AAI via his web browser. The SP requests an authentication and access control decision from the AAI consisting of at least one Identity Provider (IdP) and a Policy Decision Point (PDP). The separation is due to the advantages mentioned above. The IdP authenticates the user and requests his related attributes from all members. The authorisation request and the user attributes are transferred to the PDP. The PDP queries the SP for the resource attributes and uses the respective policies loaded at its initialisation [12]. Following to this the access decision is computed. The access control decision is forwarded to the SP via the user's browser. Complying with the idea of a generic architecture the SP enforces this decision with its own means.

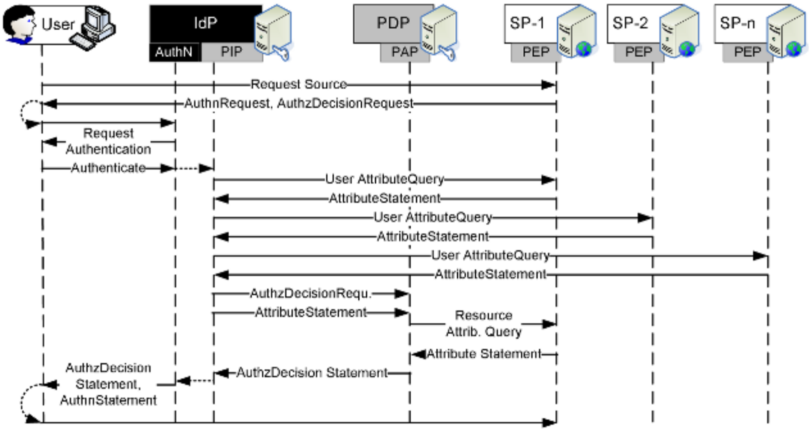


Fig. 4. Attribute-based AAI reference model

6 Conclusion and Future Work

The proposed architecture is a merger of open standards and existing functionalities, to our knowledge the first of this kind. The assessment of AAIs and classification into four levels enables the implementation of the reference with the help of existing solutions. Once enhanced with XACML technologies, Liberty's open source framework ID-FF could be a basis for such a generic solution.

We are aware of possible restrictions concerning the accumulation of user attributes. In addition, the problem of a single-point-of-failure in such a semi-centralised infrastructure needs to be discussed and evaluated. The actual implementation of such a model has to take both issues into consideration. Before considering the usage of an AAI, vendors and service providers must assess risks in comparison with their existing traditional methods. A first approach for an assessment metric can be found in [16].

An implementation of the complete architecture has yet to be built. However, single modules like SSO are realised in existing frameworks or built as proof-of-concept by the authors. See for example [14] and [2] for an attribute-based access control.

Looking back at the motivation for AAIs and ABAC, the proposed solution recommends adoption especially for e-commerce providers. To our knowledge, the given reference states a new and generic solution, meeting the criteria of a dynamic, flexible, and distributed architecture enabling the realisation of multiple synergies.

References

[1] Alfieri, R., Cecchini, R., Ciaschini, V., dell'Agnello, L., Frohner, Á., Gianoli, A., Lörentey, K., Spataro, F.: VOMS, an Authorization System for Virtual Organizations. European Across Grids Conference 2003, 33-40 (2003).

[2] Busch, S., Muschall, B., Pernul, G., Priebe, T.: Authrule: A Generic Rule-Based Authorization Module. In: Proc. of the 20th Annual IFIP WG 11.3 Working Conference on Data and Applications Security(IFIP 11.3), France (2006).

[3] Cantor, S., Kemp, J.: Liberty ID-FF Protocols and Schema Specification. http://www.projectliberty.org/specs/liberty-idff-protocols-schema-v1.2.pdf (2003).

[4] Cantor, S.: Shibboleth Architecture, Protocols and Profiles, Working Draft 05, 23 November 2004. http://shibboleth.internet2.edu/docs/draft-mace-shibboleth-arch-protocols-05.pdf (2004).

[5] Castro-Rojo, R., Lopez, D. R.: The PAPI system: point of access to providers of information. Computer Networks 37 6, 703-710 (2001).

[6] Chadwick, D. W., Otenko, A.: The PERMIS X.509 role based privilege management infrastructure. Future Generation Comp. Syst. 19 2, 277-289 (2003).

[7] Katsikas, S. K., Lopez, J., Pernul, G.: Trust, Privacy and Security in E-business: Requirements and Solutions. 10th Panhellenic Conference on Informatics (2005).

[8] Lepro, R.: Cardea: Dynamic Access Control in Distributed Systems. NAS Technical Report NAS-03-020, 1-13 (2003).

[9] Lopez, J., Oppliger, R., Pernul, G.: Authentication and Authorization Infrastructures (AAIs): A Comparative Survey. Computers & Security 23 7, 578-590 (2004).

[10] Microsoft: Microsoft.NET Passport Review Guide. www.microsoft.com/net/ services/ passport/review_guide.asp (2003).

[11] OASIS Security Services Technical Committee: Security Assertion Markup Language (SAML). http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security (2005).

[12] OASIS eXtensible Access Control Markup Language Technical Committee: eXtensible Access Control Markup Language (XACML). http://www.oasis-open.org/committees/ tc_home.php?wg_abbrev=xacml (2005).

[13] Pearlman, L., Kesselman, C., Welch, V., Foster, I., Tuecke, S.: The Community Authorization Service: Status and Future. 2003 Conference for Computing in High Energy and Nuclear Physics (2003).

[14] Priebe, T., Dobmeier, W., Kamprath, N.: Supporting Attribute-based Access Control with Ontologies. 1st International Conference on Availability, Reliability and Security (2006).

[15] Schlaeger, C., Pernul, G.: Authentication and Authorisation Infrastructures in b2c e-commerce. Proc. of the 6th International Conference on Electronic Commerce and Web Technologies - EC-Web'05, Denmark, 2005. LNCS 3590, Springer Verlag 2005.

[16] Schläger, C., Nowey, T.: Towards a Risk Management Perspective on AAIs. Proc. of the 3rd International Conference on Trust, Privacy, and Security in Digital Business - TrustBus '06, Poland, 2006. LNCS. Springer Verlag 2006.

[17] Thompson, M., Essiari, A., Mudumbai, S.: Certificate-based Authorization Policy in a PKI Environment. ACM Transactions on Information and System Security 6 4, 566-588 (2003).

[18] Welch, V., Barton, T., Keahey, K., Siebenlist, F.: Attributes, Anonymity, and Access: Shibboleth and Globus Integration to Facilitate Grid Collaboration. 4th Annual PKI R&D Workshop (2005).

[19] Yuan, E., Tong, J.: Attribute Based Access Control (ABAC) for Web Services. International Conference on Web Services 2005, 561-569 (2005).

Seamlessness and Privacy Enhanced Ubiquitous Payment

Kyoung Jun Lee[1], Mu Jeong Jeong[2], and Jeong-In Ju[3]

[1,3] School of Business, Kyung Hee University
Hoegi-dong, Dongdaemun-gu, Seoul, 130-701, Korea
{klee, jji99}@khu.ac.kr
[2] Design House, Taigwang Bldg., 162-1,
Jangchung-dong 2-ga, Jung-gu, Seoul, 100-855, Korea
coolcoom@design.co.kr

Abstract. Payment is in nature an act of money transfer from one entity to another, and it is obvious that a payment method will be valued as long as the transaction can be completed with safety no matter what technology was used. The key to U-payment is convenience and security in the transfer of financial information. The purpose of this paper is to find a desirable U-payment scheme promoting seamlessness and privacy with a strong personal device and peer-based information transactions. We propose U-SDT(Secure Direct Transfer) Protocol as a way to make transactions seamless, secure and privacy-protected.

1 Introduction

As ubiquitous computing environments evolve, business transactions are taking place more seamlessly. As a RFID tag is embedded in products, the process of entering information was replaced with just passing the RFID tag through an electromagnetic reader. Local telecommunications technologies like Blutooth and irDA incessantly send dynamic digital information to the device of others (without saving information in a USB storage device and inserting it into the counterpart's payment device to retrieve the information). Based on this technology, U-Commerce makes the real-world seamless communication of each entity's digital information possible and a seamless U-payment procedure becomes reality.

Another issue is privacy concern as mentioned heavily in many other papers on Ubiquitous computing. In some of the papers, Floerkemeier et al. (2004) proposed a RFID Protocol using "Watchdog Tag" as a way to prevent infringement of privacy. Roussos & Moussouri (2004) suggested that users in the ubiquitous computing environments should have control over their personal information through user focus group interview about MyGrocer and expressed a grave concern about exposing private information to outsiders, especially to a profit-oriented company. In addition, Acquisti(2002) explained the economic efficacy of privacy protection technologies and Langheinrich (2001) proposed to set principles of privacy protection and impose responsibility for invisible services as a way to protect privacy in the ubiquitous computing environments. Zugenmaier & Hohl (2003) emphasized the importance of keeping anonymity in the ubiquitous computing environments in order to protect user ID from being exposed to personal information collection. However, payment,

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 142 – 151, 2006.

more than any other areas, is susceptible to privacy concerns and thus merits special attention.

Cash payment is the best payment scheme to avoid privacy concerns. Many payment mechanisms were invented afterwards to enhance payment convenience, but they came at the expense of privacy. In this respect, e-cash or digital cash can be an answer to the privacy issue at a time when we have digital payment methods like e-Payment. But when we look at the evolution of e-cash, the chance is slim that e-cash becomes commonly used payment method as though the ubiquitous environment is setting foot. A future payment system will allow only publicly authorized institutions to possess minimum amount of information such as account numbers when a money transfer is made, and make it hard for merchants to collect any personal information by taking a buyer's credit card or card number to make a payment, and let buyers have control over their personal information while making a payment. This could be an alternative answer to privacy protection while condoning some involvement of a payment server for the sake of payment convenience.

In this paper, we propose a seamless U-Payment method with least privacy concern. To that end, we explain important characteristics and desirable features of ubiquitous computing environments and present a scenario in which such characteristics and features can be found and ultimately, a detailed System Architecture.

2 Characteristics of U-Payment Environment

Important characteristics of user payment environment under the ubiquitous computing environments are that creation, conversion, and transfer of payment information are made seamlessly, and functions of users' payment device, computing power and storage capacity are all very much strengthened. Such characteristics in the U-Payment environments propose a brand-new payment method to users.

2.1 Seamlessness

Seamless payment information processing simplifies payment process. For example, when you take the subway, you have had to buy a ticket and insert your ticket into the ticket slot to pass through the gate. But now, one touch of a smart card embedded with an IC chip will deliver your payment information seamlessly to the payment system of the subway. In this process, the information about cash payment is seamlessly translated into a digital form and sent to the central subway system. Even though we are using smart cards, there are some occasions that we experience some disruptions in seamless payment since the application is payee-oriented. For instance, when an elderly person or a physically challenged person tries to get a free-ride, they need ID authentication by a train officer to get a free ticket. If more seamless payment system is in place, what they have to do is just contacting the smart card embedded with the bearers' payment ID to prove they are eligible for a free-ride.

When you buy something at a local store, you will experience a similar situation. For example, when you buy an electronic gadget at a local store, if you want a money transfer via mobile, you need to enter price, account number of the merchant into the payment device and when the transfer is completed, the merchant checks the transfer was made properly and then you will get the product in your hand. On the other hand, if an RFID Tag is attached to every single product, the mobile payment device of the payer reads the information like price and account number of the payee and transfers money to the account with one touch, both the payer and the payee can complete the payment seamlessly with lower transaction costs.

Seamlessness under the U-Payment environment is a major feature that brings about changes to the user payment mechanism with regard to processing and networking of payment information. In the past U-Payment environment, conversion of the information - turning information into a digital format and vice versa - should be carried out at a high price. At local stores, information on the price and payment means are stored on a price tag or a paper manual, but in this case other substantial information remains un-encoded. However, as the information is digitalized in a seamless manner, the payment environment becomes much simpler and users are spared the hassle they had in the past and seamless payment comes into action at last.

However, the seamlessness does not mean Calm Payment. Boddupalli et al.(2003) explained that among the requirements of the U-Payment, calmness and user involvement should be balanced. Moreover, calmness should be mostly realized in low value transactions. Likewise, seamlessness of the U-Payment is not consistent with calm payment because this is just a technology-oriented payment method that fails to reflect psychological aspect of users when making a payment. When a calm payment - a payment made without a user's knowledge – is made, the user basically will not condone the fact that the payment was made without his (or her) authorization or confirmation because a payment is subtracting money from the balance of one's account and every user wants to be highly involved in the payment procedure. What we refer to as "seamlessness" here does not describe a payment made without user's consciousness but a payment made without information conversion costs.

2.2 Strong User Device

It is highly likely that individuals in the U-Commerce environments will have a personal mobile device equipped with information processing and networking functions like UDA (Ubiquitous Digital Assistant). This is the rite of path given that every transaction in the ubiquitous environments is all about information processing and networking. In particular, every individual becomes an independent commercial entity when (s)he conducts business transactions. It is a generally accepted view that people would not like the idea of incorporating such a device into a human body in the form of a microchip. Thus, chances are that a personal ubiquitous device will be a must-have item for each user.

Such a user device like UDA will perform a function as a payment device for individuals. A stronger role of a user device as a payment entity requires a more sophisticated, independent device with better information processing and networking capabilities. A user device performs the following three functions in the payment process.

Information Gathering

A payer's payment device performs a role as a seamless payment-related information reader. In the abovementioned example, when you buy an electronic gadget at a local store, a payer device reads the price and payee's bank account codified on the RFID tag. In the same way, the payer device, just like Bluetooth, will deliver seamless value to users by gathering payment-related information that is statically stored on a RFID tag and information on dynamically adjusted service charges.

Information Processing
A payer device does A to Z with regard to payment information processing. In specific, it runs a banking application, sends financial information of a buyer to the merchant's bank account after user authentication, and verifies the result of the transfer. The whole process of payment is working in a payer device.

Information Storing
Every payment-related information – during and after a transaction – will be initially stored in the Payer Device. Under the previous payment systems, payment-related information was mostly stored at a credit card company or bank that serves as a main server for the transaction. But when you use the payer device, such information is stored in the PIB (Personal Information Base) installed inside the payer device. Thus, the U-payment can be carried out while privacy of the payer is better protected.

The significance of the change in the payment scheme with the advent of strong user device can be found in the fact that payee-oriented system has given way to payer-oriented system as the main payment scheme of the ubiquitous environment. It is anticipated, as such a trend prevails, that the payer device carries out functions of both payer device and payee device, and ultimately facilitates the coming of the U-Commerce environment where each individual evolves into a business entity.

3 Suggestions of U-SDT Protocol

As described above, seamlessness and strong user device are the two important features of the U-Payment environment. Privacy protection, a thorny issue of the ubiquitous computing environments, is a critical element in the architecture of U-Payment method from the initial stage. Reflecting these factors, we propose U-SDT(Secure Direct Transaction) Protocol as a U-Payment Method which provides new value to payment entities by consolidating functions of the RFID, Payer Device, and financial institutions.

3.1 Scenario

James who works in the IT industry goes shopping in a department store to buy a present to celebrate the one year anniversary with his girl friend. James discovers a dress in the show window of a women's clothing store and goes into the store to check out the blue dress. Satisfied with the fabric and condition of the dress, James decides to purchase it. The store clerk takes the dress to the store register which reads the information included in the product tag. The product and price shows up on the monitor of the register and James has his UDA to read the payment information on the register. A payment application runs on James' UDA and James who confirms the product name, size, and price etc. on the UDA screen authenticates an official authentication. A few seconds later, the money transfer confirmation window appears on James' UDA screen from James' bank account and after the shop clerk confirms the payment through the shop's monitor on which the account confirmation window is run, he/she clicks the menu to generate a receipt. James, who thinks he might exchange it or get a refund in case his girlfriend does not like the dress, presses the reading button on the payment device and receives an electronic receipt.

The scenario above describes a form of payment focused on the buyer device in a ubiquitous environment. On the surface this is similar to the payment scenario that has been described in various papers (the payment scenario of BluePay and MyGrocer, an automatic payment process in which a mobile device of the buyer is used to recognize the product's RFID Tag) but a marked distinction exists in the flow of the payment information and its storage location and the main information processing device. In the above scenario, the biggest difference is that the processing and storage etc. of the payment information is not performed in the payee's device or server of a financial institution but is mainly performed on the payer device, which is the reason why the protection of privacy of the users is enhanced in the U-SDT.

Another important aspect that has not been revealed in the scenario is the appearance of a payment system based on the Transaction ID used in the payment transaction. This element enables payment using a financial institution without exposing the ID of the payer or payee, which was inevitable in all kinds of payments except cash payment. The generation of transaction ID is also designed not to be dependent on the existing payee device but to be a part of a system where the payer and payee mutually generate and authenticate with equal authority and unique Transaction ID is generated from the two Transaction IDs made by the payer and payee devices. A Transaction ID, which has uniqueness and representing nature, also plays an important role in the refund process. In the existing refund process, steps should be taken to confirm the breakdown of the account through the financial institution in the refund process, while when using the transaction ID all that has to be done is the refund authentication for the Transaction ID of the relevant transaction in the payee account to confirm the payment information which the Transaction ID represents in the Payment account. Such a Transaction ID plays the role of protecting privacy and enhancing efficiency of the payment.

3.2 System Architectures of U-SDT Protocol

Fig.1 shows the system architecture of the U-SDT protocol and the flows as follows:

(1) The product tag is read in or the service ID is inputted in the payee device
(2) The payee device reads in the Product ID which the payee device generates, the price amount, the encrypted payee ID (ID & account number), payee_TID
(3) After confirming the product list and price, the official authentication is confirmed (payment approval of the payer)
(4) Transport the payer_TID+payee_TID to the Payee Device and simultaneously order payment and TID to the payer account
(5) Transfer money
(6) Transport TID and payment results
(7) Confirm receipt of money
(8) Generate final TID which has the authentication of the payee's payment completion (receipt) and which the payer device reads in

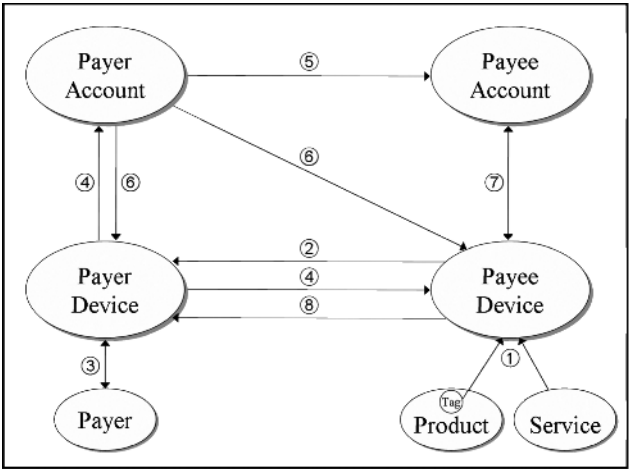


Fig. 1. U-SDT System Architecture

Payment-Related Entities

The elements involving the U-SDT payment can be divided into four categories.

First, the payer device plays the role of gathering, processing, and storing the payment-related Information. Payment-related information is read in from the payee device and the payer's Transaction ID is added to the Transaction ID which the payee device generates, creating an equal, mutual and unique Transaction ID. Through the confirmation of the official authentication with the payer, a user authentication process is created and payment process in which the actual amount of money is transferred also is processed by an application which runs on the payer device. Therefore, the payer device is the main element among the U-SDTs and possesses the largest amount of payment-related Information.

Second, the Payee device generates the initial Payment-Related Information through the product tag or the input of the service ID and generates a Transaction ID of the Payee. After payment, receipt information is generated to be transported to the Payer Device through the approval process of the Transaction ID and a signal which modifies the payment status from 'unpaid' to 'payment completed' is transported to the tag inside the product.

The third and fourth elements are the payer account and payee account which are the actual elements that transact financial information. The actual payment process is carried out between these two elements and provides value to the user with payment convenience by involving a financial server. On the other hand, since these elements perform the minimum function of exchanging financial information and do not monopolize payment-related information such as banks and credit card companies.

3.3 Structure of the Information Possession of Each Payment Entity

Another feature of the U-SDT Protocol is that it has an information possession structure in which the relevant entities possess only the essential payment information thus maximizing the protection of privacy. A payer device and payee device does not posses the other party's ID information and the payer account and payee account should not possess product list information as in Fig. 2.

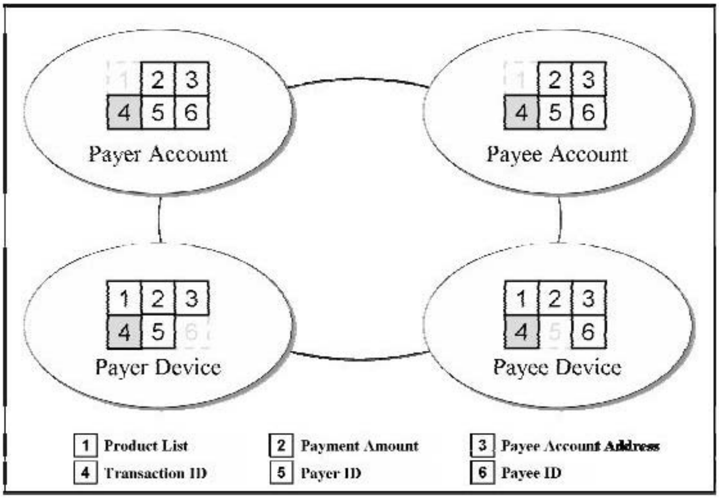


Fig. 2. The Structure of Payment-Related Information Possession in Each Entity

Classification of payment related information

The payment relevant information for the payment execution of U-SDT is as follows.

1. Product List: When possessed by a party who is not the payer the individual product name can be a serious threat to privacy. Therefore, this information should be directly possessed by those who are involved in the payment.
2. Payment Amount: Refers to the price information of a product or service. Refers to the total amount when there are a number of products and service.
3. (Encrypted) Payee Account number: The account number of the payee is the most important payment-related information required in the seamless payment process. This is encrypted and transported to protect the privacy of the payee.
4. Transaction ID: The unique Transaction ID of each transaction enables a payment and refund process to be executed using an ID for the relevant transaction without the Payer and Payee having to possess each other's ID. Such Transaction ID combines the information that is independently generated by the Payer and Payee.
5. Payer ID: Information required to confirm the payment of the payees in financial transactions between financial institutions.
6. Payee ID: Information required to confirm the payment of the payer in financial transactions between financial institutions.

The important feature of the above figure is that the payer account and payee account do not possess a product list and the payer and payee also do not possess each other's ID. A financial institution possesses the other party's ID for the transaction of financial information but since it is a third party in the payment process it does not possess information of the product list that might infringe on the privacy of the payer. In the case of buyer and seller, each financial institution that is involved in the payment and payment confirmation process may use a Transaction ID instead of exposing the IDs of the Payer and Payee to outside and prevent the leakage of privacy information such as the IDs of those participating in the transaction.

Eventually, for the value of 'seamlessness' and protection of privacy to be provided by each payment, each entity should exist in a form in which the minimum payment information essential for payment is categorized and the overall U-Payment architecture should be designed so that the payer account and payee account do not possess the product list and the payer and payee do not posses the other party's ID. Furthermore, for this to be possible, Transaction ID orientation is recommended rather than a Payer ID oriented payment.

4 Related Works

MyGrocer (Kourouthanasis et al. 2002) scenario describes the smart shopping cart automatically transporting payment information to the cashier. However, detailed information flow or system architecture is not provided. Boddupalli et al.(2003) describes a scenario of a payment system using a remote wallet and an e-tag stored in a laptop. However, it focuses on presenting requirements for a U-Payment design rather than presenting a detailed architecture. Seigneur & Jensen (2004) proposes a U-payment using anonymous digital cash stored in a mobile phone but the downside is that it is limited to incidents of small amount of payments. Gross et al. (2004) proposes a U-Payment test platform BluePay based on a PPA (Preferred Payment Architecture) and describes this using a detailed architecture and information flow. BluePay uses a device called a PTD (Personal Trusted Device) using RFID and Bluetooth technology. Payment information using short-range communication and the POS has the feature of automatically recognizing the tag of a product. In addition, it is similar to the our study that it is working on eliminating or reducing explicit interaction of the customer and that payment relevant information of the payer is stored within the PTD in the Local Exchange. However, a PTD only stores the customer ID for user authentication and payment information such as a credit card or a bank account number is stored in the backend system of a bank or a third party and user authentication is achieved through a loading method and storage by such personal information presents a possibility that privacy is infringed. As a preventive measure, our approach replaces this with an internal authentication system between a payer and payer device. Another difference is that important payments are made within POS connected to external financial data base and clients' data base and this means that an initiative of payment is heavily owned by a seller. Such a way of payment has a weakness because a seller holds a heavy amount of payment devices while a buyer gives his own financial information to a buyer to make the payment possible. Our study as an alternative proposes that an important payment should be made through the reading of payee's payment information by the payment system embedded in a payer Device.

The differences between the existing payment system and the one this study proposes are that the device of user's payment system is always reinforced, that those participants in payment own and control all information and that privacy protection is to a great extent strengthened by the use of Transaction ID and prevention of the exposure of ID.

5 Conclusions

This paper explains U-SDT protocol designed to improve privacy protection through scenario and system architecture. It also proposes a kind of structure separated from control of information regarding payment of accounts as desirable features attached to it. The key issue in this approach is to considerably decrease the high transaction costs accompanied by the conversion of offline (physical) information and digital information through the characteristic called seamlessness of ubiquitous technology. The other purpose of this study is to find a way to better protect privacy in the ubiquitous environment where it is increasingly anticipated to be infringed. Therefore, if the design of U-Payment method is practical enough to meet the two purposes, it is not only highly valued by users, but also is likely to create an independent and smart payment business model and method.

Acknowledgments

This research is supported by the Ubiquitous Autonomic Computing and Network Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

References

[1] Acquisti, A. (2002). “Protecting Privacy with Economics: Economic Incentives for Preventive Technologies in Ubiquitous Computing Environments,” *Workshop on Socially-informed Design of Privacy-enhancing Solutions, 4th International Conference on Ubiquitous Computing (UBICOMP 02).*

[2] Boddupalli, P., Al-Bin-Ali, F., Davies, N., Friday, A., Storz, O. and Wu, M. (2003) “Payment Support in Ubiquitous Computing Environments,” *IEEE Workshop on Mobile Computing Systems and Applications*, pp. 110-121.

[3] Floerkemeier, C., Schneider, R. and Langheinrich M. (2004) “Scanning with a Purpose – Supporting the Fair Information Principles in RFID Protocols,” *Institute for Pervasive Computing.*

[4] Gross, S., Fleisch, E., Lampe, M. and Müller, R. (2004). "Requirements and Technologies for Ubiquitous Payment,” *Multikonferenz Wirtschaftsinformat, Techniques and Applications for Mobile Commerce.*

[5] Kourouthanasis, P., Spinellis, D., Roussos, G. and Giaglis, G. (2002). “Intelligent cokes and diapers: MyGrocer ubiquitous computing environment,” *In First International Mobile Business Conference*, pp. 150–172.

[6] Langheinrich, M. (2001). “Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems,” *Ubicomp*, pp. 273-291.

[7] Langheinrich, M. (2002). “A Privacy Awareness System for Ubiquitous Computing Environments,” *Ubicomp*, pp. 237-245.

[8] Roussos, G. and Moussouri, T. (2004) “Consumer perceptions of privacy, security and trust in ubiquitous commerce,” *Personal and Ubiquitous Computing*, Vol. 8, No. 6, pp.416-429.

[9] Seigneur, J. and Jensen, C.D. (2004). "Trust Enhanced Ubiquitous Payment without Too Much Privacy Loss," *In Proceedings of the 19th Annual ACM Symposium on Applied Computing*, Vol. 03, pp.1593-1599.

[10] Zugenmaier, A. and Hohl, A. (2003) “Anonymity for Users of Ubiquitous Computing,” *Security-Workshop at UbiComp.*

CerTicket Solution: Safe Home-Ticketing Through Internet

F. Rico, J. Forga, E. Sanvicente, and L. de la Cruz

Department of Telematics Engineering, Polytechnic University of Catalonia
Módulo C3. Campus Nord. C/Jordi Girona 1, 08034-Barcelona, Spain
Phone: +34 93 401 60 26; Fax: +34 93 401 10 58
{f.rico, jforga, e.sanvicente, luis.delacruz}@entel.upc.es

Abstract. The Internet has fostered new ways of commerce that facilitate consumer's life, eliminating delays or displacements; electronic ticketing is one of them. Recently, several proposals have appeared that allow the procurement of tickets at the consumer's home. All these methods use either barcodes or smart cards, both of which have serious drawbacks. In fact, bar codes can be easily duplicated, and to be able to write in smart cards special hardware is needed at the user's premises. The system described in this paper solves these impediments using bar codes and smart cards jointly, in a novel way that prevents duplications without requiring additional hardware or software at the consumer's home. The verifiers located at each venue entry point are standalone devices, and are not connected in any way neither among them nor to any central database, server or portal. Nevertheless, thanks to the novel design explained in this work, all duplicate tickets will be detected, and only the valid one (whether or not it is the first to be presented at the entrance point) will be allowed to enter.

Keywords: Quality aspects in EC, E-Payment, Security and Trust, E-ticketing, contactless smart cards.

1 Introduction

Everyone in the ticketing market agrees that on line sales benefit ticketing operations by expanding the customer base and improving consumer's ease of access to tickets, as well as cutting costs through reduced staffing and hardware requirements. However, the increasing prices of tickets for some events have made counterfeiting more attractive for forgers, while cheaper, more advanced computer and copying technology have made the process of forging more accessible and simpler than ever before. Therefore, the development of new ticketing methods aimed at customers convenience could be slowed down by security concerns, which in turn have a negative impact on the revenues of ticketing companies. In fact, tickets printed at home are easily forged, and access control systems are currently unable to distinguish between genuine barcodes and copies, unless there is a central database available. Even in this case, a legitimate user can be denied access if a forged ticket is read before his. Several proposals [1][2][3] have appeared that allow the procurement of

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 152 – 161, 2006.

tickets at the consumer's home. The systems proposed in [1] and [2] guarantee the ticket authenticity, but not its uniqueness. What is more, they require a centralized data base. On the other hand, the solution presented in [3] presupposes that all the involved equipments are online when the ticket is validated.

The system described in this paper solves these problems. In fact, this system is a full proof secure method for sending and validating tickets. It uses authentication codes and portable, stand-alone verifiers working off line (i.e.: not connected in any way, neither among them nor to any central data base, server or portal). The main advantages of the system are its nil fraud possibility and its capillarity (tickets can be obtained through Internet, by fax or even by telephone). If the purchasing is done through Internet, the customer can print the ticket and its authentication code (in bar code format) directly at home. It also allows for non-assisted entry (no ticket collectors) to mass events integrating the verifier into the turnstile. In addition, as an extra value, it permits the incorporation of programs for e-coupons, points, etc, to gain customer loyalty. Customers do not need any special equipment at their premises, and the only requirement is that they are in possession of a card whose cost ranges between 1 and 2 dollars depending upon the number of cards issued.

The key idea behind the system is that the authentication code is generated for a specific card, whose active cooperation is needed to validate the ticket. Once entry to the event has been granted by the joint action of the card and the stand-alone verifier, the ticket is stored in the card. In this way the card will never validate the ticket again. Tickets are eliminated from the card memory once the event date has passed. This avoids memory starvation. The system security is granted using strong cryptographic techniques and tamper proof devices in the verifiers.

The rest of the paper is organized as follows. In the next section, the agents integrating the system are described. In section 3, we present the master lines of the system operation. The security issues are addressed in section 4. Some additional scenarios where the system could be successfully used, are explored in section 5. Finally, the conclusions appear in section 6.

2 Agents Integrating the System

Three main agents are involved in the system operation: reader operators, card issuers and portals. There may be several of each of them. A reader operator could also play the role of card issuer if so decided. In this section we analyze these agents and their interplay.

2.1 Reader Operators

Reader operators are at the system core, interfacing with both portals and card issuers (see Fig. 1). They own the readers/verifiers placed at the venue entry point.

With portals, they sign commercial agreements to sell tickets for (some of)the different events a portal offers. The event venues are labeled by the reader operator with a code, called *e* in Fig. 1.

With card issuers, reader operators cooperate carrying out a concatenated encryption scheme to be explained in the next section.

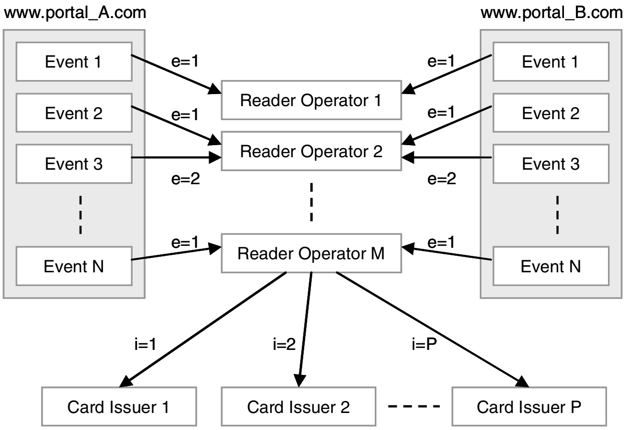


Fig. 1. System agents

2.2 Card Issuers

They are responsible for the formatting and distribution of cards to their customer base. Card issuers are uniquely identified by a number, called *i* in Fig. 1.

Customer card numbers are unique, and include a CRC (Cyclic Redundancy Check) code to detect errors. Normally, they are also prefixed by the card issuer identification number. Card numbers will be denoted by *c*.

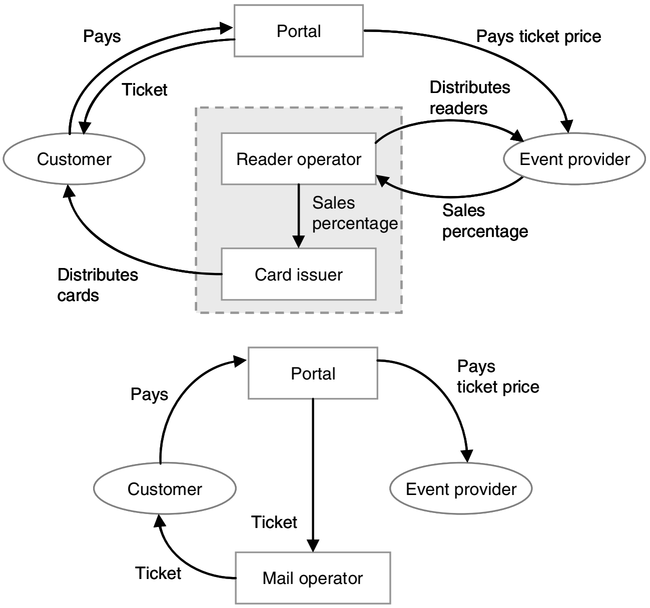


Fig. 2. Commercial interaction: Certicket solution and traditional delivery

As a result of the cooperation with the reader operator, card issuer generate the authentication code (AC) that, finally, will reach the customer, via the reader operator and the portal.

2.3 Portals

They interface with customers and readers operators. They charge customer by any of the established payment means in use today. When a customer chooses the “print at home” option, the portal asks for his/her card number and sends this information to the reader operator. Finally, the portal receives the AC from the reader operator and sends the corresponding PDF document to the customer.

In Fig. 2 we illustrate how these agents could interact commercially among them, and with customers and event providers. We also indicate the flow of money, services and goods. For comparison purposes, the nowadays more usual way of ticket delivery through a mail operator is also depicted.

3 System Operation Master Lines

In the concatenated encryption scheme previously mentioned, reader operator *r* chooses an encryption key, called internal, for each different venue, *e*. This key, denoted by $KI_{r,e}$, is stored in a protected memory area of the reader. To establish a secure session key with the card (see Sect. 3.2), the reader also has a key called master key, *KM*. This key is the same for all reader operators and it is supplied by a trusted central authority (TCA).

Likewise, card issuer *i* assigns to each card *c* a key, called external, denoted by $KE_{i,c}$. Card issuer *i* is also provided by the TCA with another key for each card, *c*. This is the card session key, KS_c, and is obtained from *KM* as $KS_c=KM(c)$.

Two phases can now be distinguished in the system operation: the document generation phase and the document validation phase.

3.1 Document Generation Phase

Once the customer has bought the ticket, and chosen the “print at home” option, the relevant data (*DD*) and card number *c* are sent to the reader operator. The reader encrypts *DD* with $KI_{r,e}$, and sends $M=KI_{r,e}\ (DD)$ and *c* to the card issuer *i* (remember that *i* is a prefix of *c*). The card issuer encrypts *M* using $KE_{i,c}$, obtaining $AC=KE_{i,c}(M)$. The code AC is now sent to the reader operator who, in turn, sends it to the portal. The portal generates the appropriate PDF document (including AC in both numeric and barcode format) that finally reaches the customer (see Fig. 3).

All the encryptions are performed using the 3-DES algorithm. If several 64 bits blocks are needed, they are encrypted in the CBC (Cipher Block Chaining) mode.

3.2 Document Validation Phase

The beneficiary handles the printed document to the validating device (verifier) and approaches his smart card to the verifier. The verifier reads the document bar code. A mutual, cryptographic strong identification between the reader and the card takes place, generating a session key (K_S), which is cooperative and random. This session key will be used to encrypt all messages between the reader and the card. The procedure to have a key K_S, common to both card and reader, is the following:

- the card sends its card number, c, to the reader,
- the reader generates a random session key, K_S,
- the reader sends $KS_c(K_S)$ to the card,
- the card obtains K_S from $KS_c(K_S)$.

Once read, the AC code is automatically sent to the smart card, that decrypts it with the key $KE_{i,c}$ (see Fig. 4). The result is sent to the verifier, which decrypts it with its own key. If the document is valid, and the card is the intended, the result will be understood by the reader. Otherwise, the result will be a gibberish random number and will therefore be rejected. Before the document is honored, the reader/writer includes it in the cancelled documents list stored in the card. A protocol is set up to guarantee the correct ending of this process. If the card detects that the document has already been validated, it informs the reader of such event so that appropriate actions can be taken. The verifier will emit a signal admitting or rejecting the document.

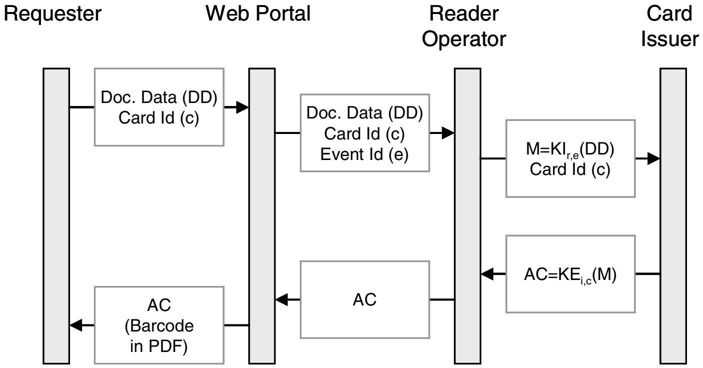

Requester
Web Portal
Reader Operator
Card Issuer
Doc. Data (DD) Card Id (c)
Doc. Data (DD) Card Id (c) Event Id (e)
M=KI$_{r,e}$(DD) Card Id (c)
AC=KE$_{i,c}$(M)
AC
AC (Barcode in PDF)

Fig. 3. Document generation phase

In this system, the smart card is the guarantor of the authenticity and unicity of the document. If the buyer duplicates it by any mean, he does not obtain any practical result since the card will not validate the document again once it has been validated, and it will show that it has been used. Therefore, to obtain any positive result from the copy, it would be necessary to duplicate the smart card, which is impossible without a huge amount of resources. On the other hand, the system allows the pre-cancellation of documents without the need of transmitting black lists to the verifiers. To cancel, the document beneficiary must go to an authorized office with the document and the card, where the document will be stored in the card as void. This way, if the buyer has kept a copy of the document, he will not be able to use it since this card will not validate it again in any case.

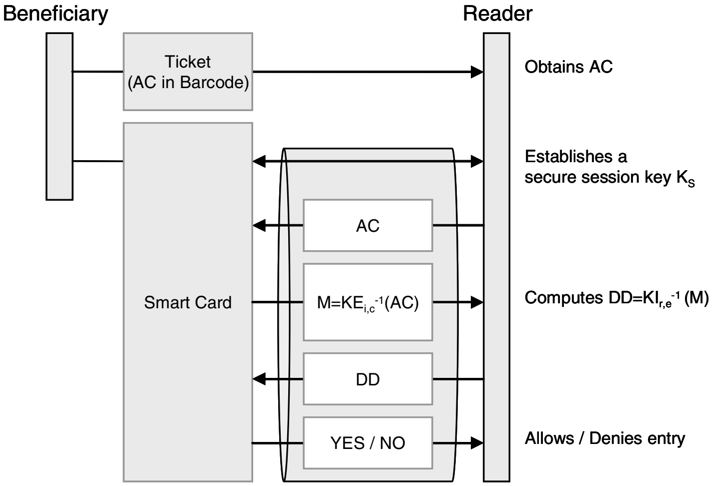


Fig. 4. Document validation phase

A few observations about the cancelled ticket list are in order:

- The documents must include an expiration date. This way they can be eliminated from the list, freeing space, once expired.
- The cards must include a residual cancellations manager, which detects cancelled documents and performs the cleaning up from a certified date provided by the verifier.
- The present date is obtained from a central server, which certifies it by means of a public key system. This certificate is passed to the card that, after checking its authenticity, erases form the list the documents that have already expired.

4 System Implementation with Protected Memory Cards

Contactless smart cards are nowadays rather expensive. Therefore to reduce the total system implementation cost, it seems advisable to use cards provided with a protected memory area, and to build up a virtual card processor in the reader.

It should be kept in mind, however, that the commercially available integrated circuits (card controllers) that implement the interface between the security microprocessor and the card, communicate in an unprotected fashion with the microprocessor. Therefore, it is necessary to incorporate a security mechanism to prevent interception and manipulation attacks to the controller-microprocessor interface [4][5]. If appropriate countermeasures were not taken, the attacker in Fig. 5 could read and write at will the content of the card memory. To protect the system against this type of attack, all the information contained in the card must be encrypted and authenticated by the security microprocessor. To that end there must be:

- Identification between controller and card by mutual challenge. As a result of this identification, the reader obtains from the card it unique card serial number,

SN_c, stored in the chip by the manufacturer. The card external key mentioned before, $KE_{i,c}$, is the concatenation of SN_c with another string, the card partial external key, $PKE_{i,c}$; i.e. $KE_{i,c}=PKE_{i,c} \mid\mid SN_c$. The card partial external key is stored in the memory encrypted with KM_i (see later).
- A block in the card memory (a counter) that can be only decremented.

It is important to notice that the previously mentioned mutual identification is needed in order to guarantee the card memory integrity. However, the encryption provided in the radio link is now superfluous since all the information stored in the card is encrypted by the security microprocessor.

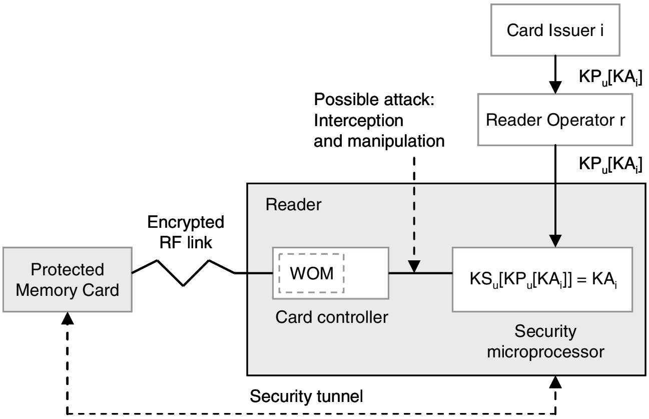


Fig. 5. Interception attack and countermeasure

In what follows, we describe an implementation we have chosen for the system that satisfies all the previously mentioned requisites.

To begin with, notice that for a reader operator to interact with a card issuer, all his readers must be in possession of the card application key, KA_i, supplied to the card issuer by the TCA. The structure and functionality of this key will be explained in the sequel.

The key is sent to the reader operator encrypted with a universal public key, KP_U. This public key is also provided by the TCA. The reader operator does not know the corresponding secret key, KS_U, and therefore is unable to obtain KAi. The reader operator is in charge of distributing $KP_U[KA_i]$ to all his readers. The reader security microprocessor obtains KA_i computing $KS_U[KP_U[KA_i]]$.

The KA_i is composed of three keys with different functionalities: Card access key (KAC_i), memory encryption key (KM_i) and memory integrity key ($KMAC_i$), i.e, $KA_i=KAC_i \mid\mid KM_i \mid\mid KMAC_i$.

The card access key is sent to the card controller WOM by the microprocessor. The other two keys never leave it.

Card and card controller authenticate mutually in several steps. The details follow:

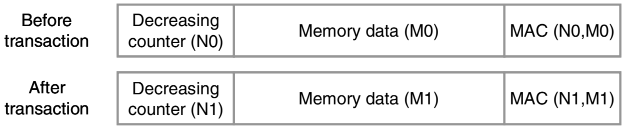
- The card sends c to the reader.
- The card controller obtains I from c. It also generates a random number R, and sends $KAC_i[R]$ to the card.

- The card obtains R and sends $KAC_i[R||SN_c]$ to the reader. The card also sends $KM_i[PKE_{i,c}]$.
- The reader obtains $R||SN_c$, authenticates the card and composes $KE_{i,c}=PKE_{i,c}||SN_c$.

The process continues now as explained in Sect. 3.

Also, in order to avoid illegitimate card manipulation, some additional actions must be carried out:

- When a card is individualized, a sector in its memory is initialized as a counter, but in such a way that it can be only decremented.
- An additional sector in the card is provided to store a memory authentication code (MAC).
- When a transaction is performed with a card, it is stored in its memory and the following actions are taken (Fig. 6): the counter is decremented, the data memory is updated and also the MAC, which authenticates both the memory and the value of the decremented counter. This way, it is not possible to manipulate the card by anyone who is not in possession of the MAC generation key, $KMAC_i$, only available to the security microprocessor.



Before transaction	Decreasing counter (N0)	Memory data (M0)	MAC (N0,M0)
After transaction	Decreasing counter (N1)	Memory data (M1)	MAC (N1,M1)

Fig. 6. Card memory operation

Proceeding in the way explained, we can make sure that the attacker in Fig. 5 cannot alter the content of the memory without being detected by a legitimate reader. In fact, this attacker will not even be able to restore the card memory to a previous state known to him (for instance: previous to the cancellation, which would allow a new utilization of the document).

5 Some Additional Application Scenarios

Although in the above paragraphs the system has been presented within the framework of the Internet and ticketing markets, it should be apparent that the PDF documents, once generated by the portal, could be sent by fax to the client if so desired. Also, the procedures described before can be used to send any kind of document with an associated monetary value to be redeemed at a remote site. Therefore, the scope of applications of the system described in this paper is very wide. This versatility will provide a positive acceptance by customers once the number of venues and events offering this system increases.

A list of possible types of documents follows:

- Movie, theater or show tickets, in which information about additional services can be printed (for instance: parking, near by restaurants, etc.).

- Tickets for trains, buses, ships or transportation in general, in all of which there is a date for the trip and a human controller.
- Plane tickets, for which a boarding card can be later obtained.
- Multi-trip cards for metropolitan transportation, such as metro, city buses, suburban trains, in which there is neither a date nor a seat reserved beforehand.
- Vouchers for hotels, festivals, or the like, in which neither date nor destiny are fixed beforehand.
- Coupons, gift certificates for department stores, etc.

The previous list is mainly focused on applications offered by private companies. However, public administrations can also take benefit from CerTicket system improving this way citizen's quality of life. For instance, Fig. 7 shows how the health services of a country could use the system to distribute medical prescriptions, eliminating unnecessary displacements.

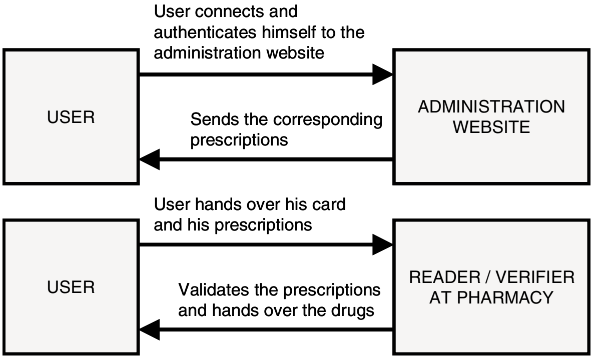

User connects and authenticates himself to the administration website
USER
Sends the corresponding prescriptions
ADMINISTRATION WEBSITE
User hands over his card and his prescriptions
USER
Validates the prescriptions and hands over the drugs
READER / VERIFIER AT PHARMACY

Fig. 7. Medical prescriptions distribution

6 Conclusion

A system has been presented that, without the need to incorporate specific peripherals in the user computers, allows them to print documents at home, guaranteeing authenticity and unicity. This opens the door to a large amount of services that, up to now, require physical displacement of users. Public administrations as well as private corporations can benefit from this system having at their disposal a mechanism that is both secure and versatile. This versatility allows the interoperation of different portals, card issuers and reader operators. It also provides anonymity, since the system does not require personal identification on part of the user. The card is impersonal and transferable, and it is not associated to any bank account, nor linked to any specific payment method.

Of course, the procedures here described have been presented only in their main features. In real practice, they are much more involved. Among other reasons, this is due to the use of contactless cards that can experience power breaches at any time if located far away from the reader. This is why precautions have been taken to guarantee that the interruption of the process at any instant does not lead, in any case, to either the loss of acquired titles or to the gain of improper rights.

Acknowledgements. This work was supported by the Spanish Research Council under projects CREDO (CICYT TIC2002-00249) and SECONNET (TSI2005-07293-C02-01).

References

1. Kobayashi, N., Bennet, Y.: System and method for delivering and examining digital tickets. Patent WO0074300
2. William, B.: Apparatus and method for issuing and validating tickets. Patent US5598477
3. Orens, G.: Method and device for delivery and use of a document opposable to third parties. Patent WO0161577
4. Anderson, R.: Security Engineering. John Wiley & Sons (2001)
5. Anderson, R., Khun, M.: Low Cost Attacks on Tamper Resistant Devices. 5th International Workshop on Security Protocols, Lecture Notes in Computer Science, Vol. 1361. Springer Verlag, London (1997) 125-136

Efficient Invocation of Web Services Using Intensional Results

Chang-Sup Park[1] and Soyeon Park[2]

[1] Department of Internet Information Engineering,
University of Suwon, Korea
park@suwon.ac.kr
[2] Department of Computer Science,
University of Illinois at Urbana-Champaign
soyeon@uiuc.edu

Abstract. Web service technologies provide a standard means for interoperation and integration of heterogeneous distributed applications on the Internet. For efficient execution of composite web services which interact hierarchically, we propose an approach to distribute invocation of web services among relevant peer systems using intensional XML data which contains external service calls and considering the costs of invocation from different peer systems. We formalize an optimization problem on the invocation of web services and provide a heuristic search method to find an optimal invocation plan and a greedy algorithm to generate an efficient solution quickly. Experimental results show that the proposed greedy algorithm can provide near-optimal solutions in an acceptable time, even for a large number of web services.

1 Introduction

Web services are emerging as a major technology for integration of heterogeneous, distributed applications on the Internet based on XML messages and Web protocols. A new application, which is a composite web service, can be easily assembled from existing web services, which supports the paradigm of Service-Oriented Architecture in software development [7]. Successive composition of web services usually creates a complex structure of interactions among a large number of distributed web services.

The concept and applications of *intensional* XML data which are XML documents containing calls to external web services have been proposed recently [3, 4, 10]. Intensional forms of data can be used to delegate invocation of web services to other web services. If a web service calls another web service with an intensional parameter, the invoked one should execute service calls embedded in the intensional data before executing its own business logic. On the other hand, a composite web service can return its caller web service an intensional result containing calls to a subset of its component web services. Then the caller may execute a portion of the delegated web service calls by itself and deliver the remainder successively to its caller through another intensional result. Fig. 1 shows an example of web service interactions using intensional results. As presented in Section 3, delegating a service call to one other web

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 162 – 172, 2006.

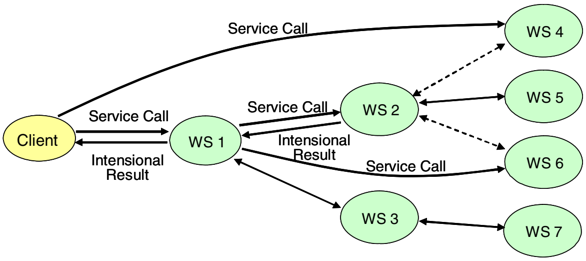


Fig. 1. An example of web service invocation using intensional results

service can affect the set of possible callers for some other web services. Thus, there exist a lot of different strategies to execute web services exchanging intensional parameters and/or intensional results.

The XML-based SOAP protocol used for invoking web services often produces considerable performance overhead on clients mainly due to XML-related tasks such as encoding, decoding, parsing, and integrating XML data [6, 8, 14]. Candidate callers for a web service typically have different resources, workloads, and communication costs with the web service. Thus, by selecting peer systems that can process service invocation tasks efficiently, we can reduce the execution costs of web services.

In this paper, we exploit intensional forms of results from web services to distribute service invocation tasks to relevant peer nodes aiming at improving performance of the overall web service systems. After briefly discussing previous work on web services and intensional XML data in Section 2, we formally describe an optimal-cost invocation plan for hierarchically interacting web services in Section 3. Then we propose heuristic algorithms based on the A* search and greedy method to generate either an optimal invocation plan or an efficient solution in Section 4. We present experimental results on the effectiveness and performance of the proposed methods in Section 5 and draw a conclusion in Section 6.

2 Related Work

The deployment of web services is supported by various standards including Web Service Description Language (WSDL), Universal Description, Discovery, and Integration (UDDI), and Simple Object Access Protocol (SOAP). They respectively support definition of the interfaces of web services, advertisement of web services to the community of potential user applications, and binding for remote invocation of web services [15]. Recent researches on web services include automatic composition, quality improvement, and semantic description and discovery of the web services [5].

Recently, several approaches and applications have been proposed which exploit intensional data embedding calls to external functions or web services [3, 4, 10]. The Active XML (AXML) [1] provides a declarative framework for data integration and management on the web utilizing web services and intensional XML data, called the AXML documents, in a peer-to-peer environment. Each peer system provides AXML services which allow clients to access AXML documents stored in the peer's repository. Defined as parameterized XML queries over the AXML documents, they can be used to search, gather, and integrate data distributed over the peer systems.

Related with the intensional data, S. Abiteboul, *et al.* [2] studied on effective distribution and replication of XML data and web services, and T. Milo, *et al.* [11] provided document and schema rewriting algorithms to support exchange of intensional data between heterogeneous applications. N. Ruberg, *et al.* [14] have suggested an approach to optimize materialization of AXML documents in P2P environments. Assuming that peer systems provide a generic query service which can execute services calls offered by any other peers, it proposed some heuristics to find an efficient distributed materialization strategy. However, intensional results were not considered in executing AXML services, and detailed algorithms and performance evaluations for the proposed methods were not presented in the work.

3 Problem Definition

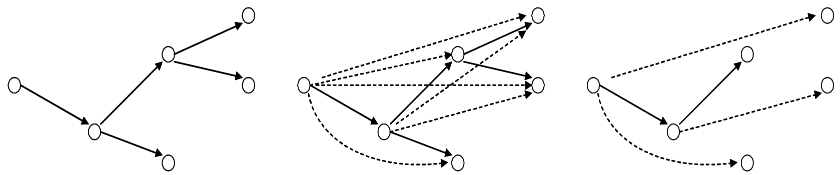
In this section, we formalize the concept of an optimal-cost invocation plan for web services using intensional results. For optimization of global invocation strategies, we assume that access and composition information are given from web service providers through a registry service system such as UDDI.

First, we describe a cost model for web service invocation briefly. Executing a service call generally requires client-side tasks, server-side tasks, and transmission of input and output messages between the client and server. The client-side tasks include initializing service activation, generating an input SOAP message, decoding and parsing a result SOAP message, and integrating the results from multiple web service calls embedded in an intensional result, if any. Not considering replication of web services, the cost of server-side tasks is regardless of the client invoking the web service. Thus, we define the cost of invocation of a web service as the sum of the client-side cost and the communication cost.

For cost-based optimization, we need to estimate the above-mentioned cost factors. Estimation of client-side costs can be achieved by monitoring the workload and performance of each node in real-time. Communication cost between a pair of nodes can be estimated by measuring network traffic and communication latency between them [9, 14].

For the simplicity of description, we assume that there exists at most one sequence of service calls between any pair of web services and there is no cycle in it, as shown in Fig. 1. Then, interactions of the web services can be represented as a directed tree rooted at the client.

Definition 1. (Web Service Call Definition Tree) $DT = (V_d, A_d, W_d)$ is a weighted directed tree which represents the call relations among web services, where V_d is the set of finite number of vertices representing web services, A_d, a subset of $V_d \times V_d$, is the set of arcs denoting service calls and $W_d{:}A_d \rightarrow Z^+$ is a function that defines a service invocation cost for each arc in A_d. The path $(v, s_1, s_2, \ldots, s_n, w)$ from a node v to a node w in DT is called the *Call Definition Path* from v to w and denoted by $P_d(v, w)$.



(a) Call definition tree DT (b) DT^* (c) Invocation tree ET

Fig. 2. An example of a call definition tree and an invocation tree for web services

If a call definition path $P_d(v, w) = (v, s_1, s_2, \ldots, s_n, w)$ exists in DT for two web services v and w, an invocation instance is possible in which intensional results embedding a service call to w are transmitted from s_n to v through the web services in $P_d(v, w)$ in the reverse order and v directly invokes w. Thus, for a given $DT = (V_d, A_d, W_d)$, a graph $DT^* = (V_d, A_d^*, W_d^*)$ where A_d^* is the transitive closure of A_d and $W_d^*:A_d^* \rightarrow Z^+$ is a cost function on A_d^* satisfying $W_d \subseteq W_d^*$ represents all the possible ways of invoking the web services in V_d by exploiting intensional results (See Fig. 2-(a) and 2-(b)).

Definition 2. (Web Service Invocation Tree) For a given call definition tree $DT = (V_d, A_d, W_d)$, $ET = (V_e, A_e, W_e)$ is a weighted directed tree which represents a feasible instance of invoking web services using intensional result, where $V_e = V_d$, $A_e \subseteq A_d^*$, and $W_e:A_e \rightarrow Z^+$ satisfying $W_e \subseteq W_d^*$. The path $(v, s_1, s_2, \ldots, s_n, w)$ from a node v to a node w in ET is called the *Invocation Path* from v to w and denoted by $P_e(v, w)$.

In any invocation tree ET, no pair of vertices s_i and s_j can exists such that both of arcs (s_i, s_k) and (s_j, s_k) exist at the same time for a vertex s_k, since every web service must be invoked once from a single client by the definition of DT. Therefore, invocation tree ET must be a directed spanning tree for DT^* (See Fig. 2-(c)). Every directed spanning tree for DT^*, however, does not mean a feasible invocation instance for web services. Invocation trees for DT have the following properties.

Lemma 1. If there is a call definition path $P_d(v, w)$ for a pair of web services v and w in DT, there exists no invocation tree for DT containing an invocation path $P_e(w, v)$.

Lemma 2. Assume that DT has a call definition path $P_d(v, w) = (v, s_1, s_2, \ldots, s_n, w)$ ($n \geq 1$) for a pair of web services v and w. If an invocation tree for DT has an arc (v, w), it also has an invocation path $P_e(v, s_i)$ for all the web services s_i ($1 \leq i \leq n$).

We omit the proofs of the above lemmas for space limitation. They can be proved easily by definition of the invocation tree and by induction for the length of the call definition path $P_d(v, w)$. From the lemmas, we have the following result (See Fig. 3).

Theorem 1. If DT has a call definition path $P_d(v, w) = (s_1, s_2, \ldots, s_n)$ ($s_1 = v$, $s_n = w$, $n \geq 4$) for a pair of web services v and w, there is no invocation tree for DT containing two arcs (s_i, s_k) and (s_j, s_m) where $1 \leq i < j < k < m \leq n$.

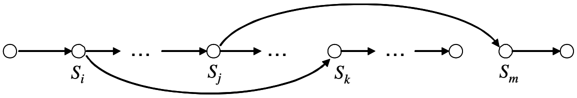


Fig. 3. An infeasible invocation plan using intensional results

(*Proof*) Assume that $(s_i, s_k) \in A_e$ and $(s_j, s_m) \in A_e$ for an invocation tree $ET = (V_e, A_e, W_e)$. Since $(s_j, s_m) \in A_e$ and $j < k < m$, there exists an invocation path $P_e(s_j, s_k)$ in ET by Lemma 2. However, since $P_e(s_j, s_i)$ is not in ET by Lemma 1, s_i cannot be contained in $P_e(s_j, s_k)$. Thus, s_k should be called from s_i as well as from another web service, but this cannot be a legal execution instance of web services using intensional results. Therefore, ET cannot have both of (s_i, s_k) and (s_j, s_m) in A_e. □

Considering the above theorem, we formalize optimization of the invocation strategies of web services delivering intensional result.

Definition 3. (Optimal Invocation Plan) For a given call definition tree $DT = (V_d, A_d, W_d)$ for a set of web services, consider the set Π defined by,

Π = {T | T is a directed spanning tree for DT^* which has no pair of arcs (s_i, s_k) and (s_j, s_m) $(i < j < k < m)$ for any call definition path $P_d(s_i, s_m) = (s_i, s_{i+1}, \ldots, s_m)$ in DT}.

We call the tree in Π an *invocation plan* for DT and the tree $T_0 \in \Pi$ satisfying $Weight(T_0) = \underset{T \in \Pi}{Min}\{Weight(T)\}$ the *optimal invocation plan* for the given web services exchanging intensional results, where $Weight(T) = \sum_{a \in A} W(a)$ for a tree $T = (V, A, W)$.

4 Optimization

4.1 Exhaustive Search

A simple approach for optimization is to search the solution space exhaustively and select the invocation plan having a minimum execution cost. We can enumerate all the possible invocation plans systematically by considering the vertices in the given call definition tree in the increasing order of their depth. Denoting the set of invocation trees for the subset of vertices whose depths are no deeper than k by $E(k)$, we can generate a subset of invocation trees in $E(k)$ from each tree in $E(k\text{-}1)$ by selecting a set of caller vertices for the vertices of depth k in DT and adding new arcs.

While the exhaustive search always finds the optimal solution, it requires huge amount of execution time. Assuming that DT is a perfect tree in which all internal nodes have the same out-degree f and all leaf nodes have the same depth h, we have

$$|E(h)| = \sum_{k_1, k_2, \ldots, k_{h-1}} \binom{f^{h-1}}{k_1, k_2, \ldots, k_{h-1}} (2^f)^{k_1} \cdot (3^f)^{k_2} \cdot \cdots \cdot (h^f)^{k_{h-1}} = (2^f + 3^f + \cdots + h^f)^{f^{h-1}} = \left(\sum_{i=2}^{h} i^f\right)^{f^{h-1}}$$

where $k_1 + k_2 + \ldots + k_{h-1} = f^{h-1}$ and $h \geq 2$.

It means that as the fan-out and height of the call definition tree increase, the number of possible invocation trees grows drastically. For example, even for the small

```
1  **A* Search Algorithm**
2  **Input:** a web service call definition tree $DT = (V_d, A_d, W_d)$
3  **Output:** an optimal-cost invocation tree for $DT$
4  **begin**
5      Let the start state $s_0 = ((\{r\}, \varnothing, \varnothing), 0)$.
6      Let $OPEN$ be a priority queue storing states in a non-decreasing order of
       values of $f()$ defined in equation (1).
7      $OPEN := \{ s_0 \}$;
8      **loop**
9          Select from $OPEN$ a state $s$ having the smallest value of $f(s)$.
10         $OPEN := OPEN - \{ s \}$;
11         Let the selected state $s = (ET_s, d_s)$ where $ET_s = (V_s, A_s, W_s)$.
12         **if** $V_s = V_d$ **then return** $ET_s$; **end if**;
13         $N := \{ w \mid w \in V_d - V_s$ and its depth in $DT$ is $d_s+1 \}$;
14         **for** each possible invocation tree $ET_n = (V_n, A_n, W_n)$, where $V_n = V_s \cup N$,
               $A_n = A_s \cup \{ (u, w) \mid$ for each $w \in N$, $u$ is a node in $P_e(r, v)$ in $ET_s$ where
               $v$ is a node such that $(v, w) \in A_d \}$, and $W_n{:}A_n \rightarrow Z^+$ satisfying $W_s \subseteq W_n \subseteq W_d^*$
15         **do**
16             Generate a successor state of $s$, $s_{succ} = (ET_n, d_s+1)$.
17             $OPEN := OPEN \cup \{ s_{succ} \}$;
18         **end for;**
19     **end loop;**
20 **end.**
```

Fig. 4. A* search algorithm

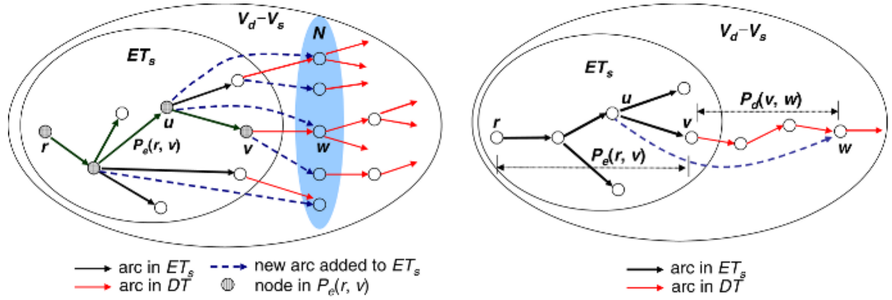
values of $f = 3$ and $h = 4$, the number of invocation trees amounts to $99^{27} = 7.6\text{e}{+}53$. In terms of the number n of web services, the method has time complexity of $O(h^{2f^h}) = O(\log^{2n} n)$, hence it is hard to be used in real environments.

4.2 A* Search Algorithm

In this section, we provide an A* algorithm for finding an optimal invocation plan as shown in Fig. 4. A* algorithms search for an optimal solution in an implicit state-space graph heuristically, avoiding an exhaustive search by pruning a part of the search space [12].

Given $DT = (V_d, A_d, W_d)$ where the root vertex is denoted by r, a state s in the search space represents an invocation tree ET_s for a sub-graph of DT containing all the vertices whose depth is no deeper than d_s in DT. The A* algorithm begins with a start state s_0 for r in the set $OPEN$ of active states to be explored. At each stage, it selects a state s having the smallest value of $f(s)$, which is a heuristic estimation of the cost $f^*(s)$ of an optimal solution that can be achieved from s, *i.e.*, a minimum cost invocation tree for DT containing ET_s as a sub-graph. Then, as shown in line 10~14 in Fig. 4, it expands s by generating its successor states representing invocation trees which have ET_s as a sub-graph and additionally contain the arcs from the nodes in ET_s to the nodes of depth d_s+1 in DT (Refer to Fig. 5-(a)).

$f^*(s)$ and $f(s)$ can be expressed as $f^*(s) = g^*(s) + h^*(s)$ and $f(s) = g^*(s) + h(s)$, respectively, where $g^*(s)$ denotes the cost of ET_s. With $h(s)$ satisfying $h(s) \leq h^*(s)$ for all



(a) Generating a successor state of *s* (b) Computing the heuristic function of *s*

Fig. 5. Execution of the A* algorithm

states *s*, A* algorithms guarantee to terminate with an optimal-cost solution [12]. In order to estimate $h^*(s)$ optimistically, we consider an auxiliary tree HT_s for $ET_s = (V_s, A_s, W_s)$, defined by

$HT_s = (V, A, W)$ where $V = V_d$, $A = A_s \cup \{(u, w) \mid w \in V_d - V_s$, u is a node in $P_e(r, v) \cdot P_d(v, w)$ such that $W_d^*((u, w))$ is minimal, where v is a leaf node in ET_s which is contained in $P_d(r, w)\}$, and $W{:}A \rightarrow Z^+$ satisfying $W_s \subseteq W \subseteq W_d^*$.

As a spanning tree for *DT* containing ET_s as a sub-graph, HT_s includes an incoming arc for each of the vertices not in ET_s but in *DT*, which has a minimum invocation cost regardless of other vertices (See Fig. 5-(b)). Thus, it may not be a feasible invocation plan for *DT* satisfying Theorem 1 while we have $Weight(HT_s) \leq Weight(ET)$ for all the invocation trees *ET* for *DT* which contain ET_s. Therefore, if we define

$$f(s) = Weight(HT_s) = Weight(ET_s) + \sum_{w \in V_d - V_s} \min_{u \in P_e(r,v) \cdot P_d(v,w)} W_d^*((u,w)) \qquad (1)$$

where v is a leaf node in ET_s which is contained in $P_d(r, w)$, we have $f(s) \leq f^*(s)$ and hence $h(s) = f(s) - g^*(s) \leq f^*(s) - g^*(s) = h^*(s)$. Consequently, the proposed algorithm guarantees the generation of an optimal-cost invocation plan for given web services.

4.3 Greedy Algorithm

The A* algorithm proposed in Section 4.2, though being able to find an optimal solution, may experience significant performance degradation as the number of web services increases. In this section, we propose a greedy algorithm to produce a cost-effective invocation plan in a more efficient way.

As shown in Fig. 6, the algorithm traverses the given *DT* in a breadth-first manner to build an invocation tree *ET*. For each vertex *w* in *DT* visited during search, the caller vertex *u* of *w* is selected among the vertices already in *ET* and the new arc (*u*, *w*) is inserted into *ET*. We select *u* from the ancestors of *w* which are on the invocation path $P_e(r, v)$ from the root *r* to the parent *v* of *w* in *ET*, considering not only the cost of invoking *w* but also the invocation costs for descendent web services of *w*. Specifically, denoting the set of descendents of *w* by *Desc*(*w*), we select the vertex *u*

```
Greedy Algorithm
Input: a web service call definition tree $DT = (V_d, A_d, W_d)$
Output: an invocation tree for $DT$
begin
    Let $Q$ be a queue to store nodes in $DT$. $Q := \emptyset$;
    Let $ET = (V_e, A_e, W_e)$ be the result invocation tree. $ET := (\{r\}, \emptyset, \emptyset)$;
    while $Q \neq \emptyset$ do
        $v :=$ Dequeue($Q$);
        Let $(s_1, s_2, \ldots, s_n)$ be the sequence of nodes in $P_e(r, v)$ in $ET$.
        for each child node $w$ of $v$ in $DT$ do
            Find $u$ such that $u = \underset{u' \in P_e(r,v)}{Min} \left( W_d^*((u', w)) + \sum_{x \in Desc(w)} \underset{t \in P_e(r,u')}{Min} W_d^*((t, x)) \right)$.
            $V_e := V_e \cup \{w\}$; $A_e := A_e \cup \{(u, w)\}$; $W_e := W_e \cup \{W_d^*((u, w))\}$;
            Enqueue($Q$, $w$).
        end for;
    end while;
    return $ET$;
end.
```

Fig. 6. Greedy algorithm

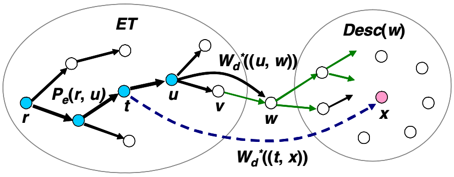


Fig. 7. Selecting the caller of w in the greedy algorithm

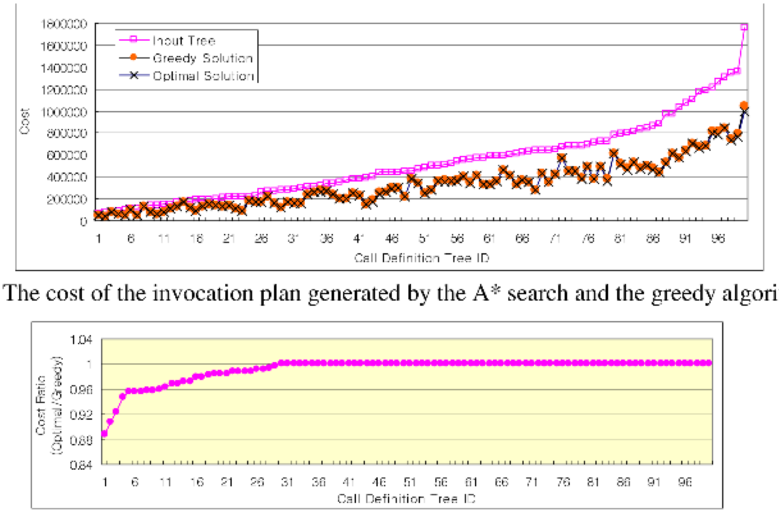
in $P_e(r, v)$ having the minimal value of $W_d^*((u,w)) + \sum_{x \in Desc(w)} \underset{t \in P_e(r,u)}{Min} W_d^*((t,x))$ as the caller of w (See Fig. 7). Note that when a web service u invokes w, the descendents of w can be called only from the vertices in $P_e(r, u)$. This requires that for the vertices in $Desc(w)$ we should find their callers with minimum costs repetitively from the different sets of candidates determined by the position of the caller u of w on the path $P_e(r, v)$. Therefore, time complexity of finding the caller of w is $O(|Desc(w)| \cdot l^2)$ where l is the length of $P_e(r, v)$. It can be improved to $O(|Desc(w)| \cdot l)$ by using extra memory space in $O(|Desc(w)|)$. Specifically, we can avoid a lot of cost comparisons by considering the candidate caller u of w on $P_e(r, v)$ in the increasing order of depth, storing for each vertex x in $Desc(w)$ the weight of the arc (t, x) from a caller t in $P_e(r, u)$ to x which has a minimum invocation cost for the position of u, and reusing the stored information in the next stage with the next position of u. The extended algorithm is shown in [13]. As a result, for a call definition tree of n vertices which is a perfect tree as in Section 4.1, the greedy optimization algorithm can be executed in

$$O(\textstyle\sum_{i=1}^{h} |Desc(w)| \cdot i \cdot f^i) = O(\textstyle\sum_{i=1}^{h} \frac{f^{h-i+1} - 1}{f - 1} \cdot i \cdot f^i) = O(\frac{f}{f-1} n \log_f{}^2 n - 2n \log_f n) = O(n \log^2 n).$$

5 Performance Evaluation

This section evaluates effectiveness and performance of the optimization methods proposed in Section 4. By experiments using test call definition trees of web services, we assessed qualities of the invocation plans generated by the proposed greedy method and A* search algorithm, which were compared with the costs of those test DTs. We also measured and compared the execution time taken by the algorithms.

We implemented the proposed algorithms in C++ with the standard template library and conducted experiments on a server employing an Intel Xeon 3.2GHz processor and 6GB main memory and running the Red Hat Linux 9.0 operating system. In the experiments, 100 test call definition trees of web services were used which have the height of 3 or 4 and the fan-out from 0 to 3, selected at random. Invocation costs of web services were also randomly defined as a value between 1 and 65535.



(a) The cost of the invocation plan generated by the A* search and the greedy algorithm

(b) The cost ratio of the optimal solution to the greedy solution

Fig. 8. Quality of the optimal solutions and greedy solutions

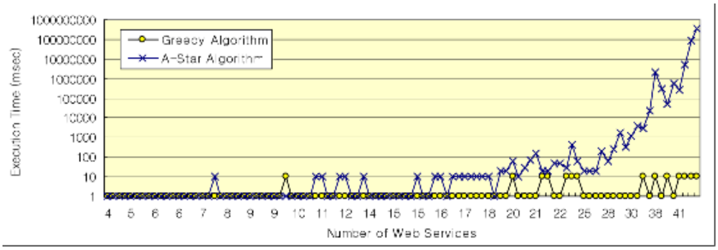


Fig. 9. Execution time of the A* search and the greedy algorithm

Fig. 8-(a) shows the costs of the invocation plans produced by the A* algorithm and greedy algorithm for the test call definition trees, arranged in a non-decreasing order of the costs of the input DTs. The result indicates that the costs of the optimal and greedy solutions are respectively 61.7% and 62.4% of the cost of the test DTs on the average. The cost reductions are the result of efficient distribution of invocation tasks for web services and tend to improve as the height and fan-out of call definition trees increase, as shown in [13]. Fig. 8-(b) presents the quality of the greedy solution measured by the cost ratio of the optimal solution to the greedy one. We observe that most of the solutions obtained by the proposed greedy algorithm are very close to optimal. It yielded an optimal solution for 71% of the test DTs and a sub-optimal solution had the quality of around 96.8% of that of the optimal solution on the average.

Fig. 9 shows the CPU time taken by the greedy and A* algorithm. The results are arranged in a non-decreasing order of the number of web services in the test DTs. The greedy algorithm completed in less than 10 msec for all the test DTs while the A* algorithm took a larger amount of time with a maximum of about 93 hours. Moreover, the execution time of the A* algorithm severely increases as the number of web services grows. We observe that the greedy algorithm scales up for a large and complex structure of composite web services well in contrast to the A* algorithm.

6 Conclusion

In this work, we proposed a cost-based optimization method to execute invocation of composite web services efficiently by distributing the tasks involved in activating the web services over the peer systems using intensional data. We formalized the optimal invocation plan for web services which interact hierarchically and deliver intensional results. We analyzed difficulties of the exhaustive search in real environments and provided an A* heuristic algorithm to find an optimal solution. We also suggested a greedy algorithm to generate an efficient solution quickly. We showed by experiments that the proposed method can enhance the overall performance of web services by providing an efficient invocation plan close to the optimal one and has good scalability for the number of web services.

References

1. Abiteboule, S., *et al.*: Active XML: A Data-Centric Perspective on Web Services. Technical Report, No.381. GEMO, INRIA Futurs (2004)
2. Abiteboule, S., *et al.*: Dynamic XML Documents with Distribution and Replication. Proc. of ACM SIGMOD (2003)
3. Active XML. http://activexml.net/
4. Apache Jelly: Executable XML. http://jakarta.apache.org/commons/jelly/
5. Curbera, F., *et al.*: The Next Step in Web Services. CACM, 46(10) (2003) 29-34
6. Davis, D., Parashar, M.: Latency Performance of SOAP Implementations. Proc. of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid2002) (2002) 407-412
7. Erl, T.: Service-Oriented Architecture: Concepts, Technology, and Design. Prentice Hall (2005)
8. Kohlhoff, C., Steele, R.: Evaluating SOAP for High Performance Business Applications: Real-Time Trading Systems. Proc. of WWW '03 (2003)

9. Liu, Y., Ngu, A., Zeng, L.: QoS Computation and Polishing in Dynamic Web Service Selection. Proc. of WWW '04 (2004)
10. Macromedia Coldfusion MX. http://www.macromedia.com/
11. Milo, T., Abiteboul, S., Amann, B., Benjelloun, O., Dang Ngoc, F.: Exchanging Intensional XML Data. Proc. of ACM SIGMOD Conference (2003)
12. Nilsson, N. J.: Artificial Intelligence: A New Synthesis. Morgan Kaufmann Publishers, Inc., San Francisco, CA (1998)
13. Park, C. S., Park, S.: Distributed Execution of Web Services Using Intensional Data. Technical Report, Dept. of Internet Information Engineering, University of Suwon (2006)
14. Ruberg, N., et al.: Towards Cost-based Optimization for Data-intensive Web Service Computations. Proc. of Brazilian Symposium on Databases (2004)
15. Tsalgatidou, A., Pilioura, T.: An Overview of Standards and Related Technology in Web Services. Distributed and Parallel Databases, 12(2) (2002) 135-162

On Distributed Service Selection for QoS Driven Service Composition*

Fei Li, Fangchun Yang, and Sen Su

State Key Lab. of Networking and Switching, Beijing University of
Posts and Telecommunications
187#, 10 Xi Tu Cheng Rd., Beijing, 100876, P.R.China
pathos.lf@gmail.com, {fcyang, susen}@bupt.edu.cn

Abstract. The service oriented paradigm promises dynamic service discovery and on-the-fly service composition to fulfill user requirements. For the dynamic and heterogeneous nature of service and network, ensuring end-to-end QoS (Quality of Service) becomes a challenge in service composition. This paper proposes a distributed QoS registry architecture, in which QoS registries could maintain services cooperatively and exchange information. Then we present a model combining network QoS and service QoS together to show their impact on end-to-end QoS. Based on the distributed QoS registry and combined QoS model, we present a distributed service selection algorithm to find optimal service composition plan. Experimental results show that the algorithm achieves excellent selection result with outstanding performance in our architecture.

1 Introduction

Web Service is an emerging technology aiming at effectively integrating and reusing business applications in global scale. In web service framework, services could describe and publish their functional and non-functional properties in standard way. Customers could find the right service to fulfill their requirements through service discovery process. Service composition mechanism enables individual services to compose together to provide end user a more powerful service (composition service), which greatly improves service reuse [10].

Automatic service composition is an important issue in web service research. In the past, large amount of works have been done on validation of composition service logic, semantic discovery of appropriate services and monitoring of service execution [11]. Recently, pioneer researchers have proposed a new question in service composition: How to select a set of appropriate services to instantiate composition logic and satisfy user's QoS requirements? This is called a *quality driven service*

* This work is supported by the National Basic Research and Development Program (973 program) of China under Grant No.2003CB314806; the Program for New Century Excellent Talents in University (No: NCET-05-0114); the program for Changjiang Scholars and Innovative Research Team in University (PCSIRT).

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 173 – 182, 2006.

composition problem [1]. Some early works have presented basic QoS models for composition services. In these models, a QoS center/registry is often responsible for maintaining QoS information and carrying out the computation. The common accepted QoS criteria are execution duration, price, availability and reliability.

However, 2 problems may hinder the application of these early solutions: Firstly, they assume only one QoS registry maintains all the geographical distributed autonomous services which is not practical in real world. As a result, all the selection algorithms proposed could only execute in centralized mode. Secondly, most of the works focus on service QoS and its compositional effect. For efficiently composing services on the internet, the network conditions also has effect on compositional QoS, e.g. network delay between services, bandwidth between services and network availability between services. Especially, for interactive applications or network intensive applications, e.g. online game, interactive multimedia system, the real-time network conditions and service conditions both have great impact on end-to-end QoS [6]. To address these problems, we proposed a distributed QoS registry architecture at first. The basic idea of our QoS registry is, every registry maintains a set of services, and registries are interoperable. Then we present a general QoS model for composition service including network QoS and service QoS. To effectively integrate the model into service composition process, we propose a heuristic service selection algorithm based on Extended Bellman-Ford Algorithm (EBFA) [2]. EBFA is a well-known algorithm in QoS routing to solve *multi-constrained path routing problem.*

The rest of the paper is organized as follows: Section 2 reviews some related works. Section 3 describes the distributed QoS registry architecture. Section 4 presents our QoS model in detail. Section 5 presents the service selection algorithm and its optimization. Section 6 discusses the algorithm and its complexity in different cases. Finally, the paper concludes in Section 7.

2 Related Works

Traditional Web Service QoS registrys focus on gathering QoS information from services and evaluating services. Serhani et al. [12] have presented a QoS architecture and its relationship with other entities in Web Service architecture. Maximilien et al. [13] focused on the policy architecture of processing QoS information. As far as we know, no works has mentioned the requirements of interoperation between registries.

QoS issue in service composition is receiving more and more attention from both academia and industries. Zeng et al. [3] has presented a basic QoS model for service composition. Although they model only 5 most common QoS criteria: Response time, cost, availability, reliability and reputation, the model has very good extensibility for business service QoS. However, the Integer Programming computation for QoS may suffer from its complexity and could only execute in centralized mode.

Gu et al. [6][7] have presented a model similar to ours in their multimedia service composition research. Their model contains link delay and link availability but the model could not reflect the complexity of composition service QoS in web service. Based on their model, they use an optimized Dijkstra algorithm to solve the problem, which is inefficient in large network. Dijkstra algorithm also requires the execution entity to have global view of the whole network.

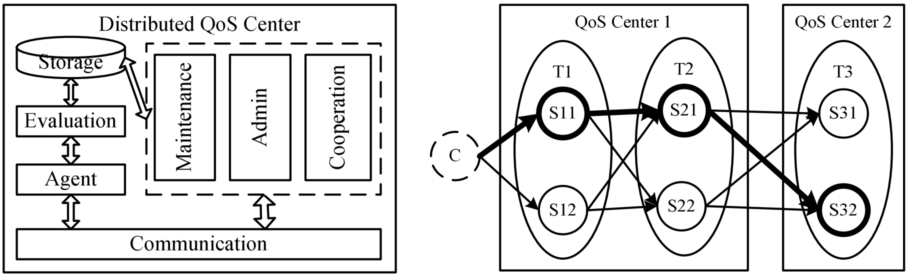
Because the service selection problem is NP-hard [4], many heuristic approaches have been carried out to improve efficiency. Canfora et al. [4] proposed a genetic algorithm approach to optimize the selection process and gave a comprehensive comparison to Integer Programming approach. Yu et al. [5] presented 2 models for the problem. One is a combinatorial model which defines the problem as the Multi-dimension Multi-choice 0-1 Knapsack Problem (MMKP) and solved by modified HEU heuristic algorithm. The other is a graph model which defines the problem as the Multi-Constraint Optimal Path (MCOP) problem and solved by modified Multi-Constraint Shortest Path (MCSP) algorithm. Their approach shows outstanding performance in experiment, but still could only execute in centralized mode.

3 Distributed QoS Registry Architecture

In service composition process, there are 2 important entities responsible for 2 phases respectively: *composition engine* and *QoS registry*. Composition engine could generate and validate composition logic, the output is represented as task flow. A task flow is an abstract logic to fulfill user's functional requirement. We have to select one service instance for each task in the flow. This stage is accomplished by the QoS registry.

In our distributed QoS registry architecture, a registry could interoperate with other registries. The interoperation functions are implemented in standard Web Service interface. We currently implemented 3 kinds of protocols: The first kind is *Maintenance*, including registry status report, load report, restart and shutdown registries. The second is *Administration*, e.g. transferring a service's registration to other registries based on service request. The third is *Cooperation*. The major usage of this kind is for planning service composition, described in section 5. The architecture illustrated in Fig.1(a). The *Agent* is responsible for collecting information from services. All interactions handled by *Communication* module. A QoS registry maintains at least 1 set of functionally identical services and different registry could only maintain functionally different services' QoS information. To include network condition in service selection process, our QoS registry maintains link information between services. The link information could be probed by client-side application and processed by QoS registry with history records before selection [8][9]. Here, the word "client" stands for both the end-user application and the middle services which may act as client from the viewpoint of successive services in composition logic.

User experienced QoS largely depends on one task path which has the greatest impact on the user interested QoS criteria, we call this task path a *critical task path (CTP)*, which could be predicted by composition engine with history data. A critical task path with n tasks is: $CTP = \langle t_1, t_2, ...t_n \rangle$, where $t_i \left(1 \le i \le n\right)$ is the ith task in topological order. Each task has a set of candidate services, $S\left(t\right) = \left\{s_1, s_2, ...s_m\right\}$, where m is the number of services which could fulfill task t . Each service has a set of non-functional properties. In composition service execution, candidate services from different tasks bind together to fulfill user requirements. Thus there is a virtual link between every pair of candidate services from adjacent tasks. A *service path* is represented as $SP = \left\langle l_1, s_1, l_2, s_2, ...l_n, s_n \right\rangle$, where $s_i (1 \le i \le n)$ is the service selected for task t_i , $l_i \left(2 \le i \le n\right)$ is the link between s_i and s_{i-1} , and l_1 is the link between client and s_1 . Fig.1(b) illustrated a service path across multiple QoS registries, $\left\langle t_1, t_2, t_3 \right\rangle$ is a critical task path.



(a)

(b)

Fig. 1. (a) Distributed QoS registry Architecture; (b) Service Path across different QoS registry

4 QoS Model for Composition Services

The QoS model contains two types of basic QoS criteria: Service related QoS criteria and link related QoS criteria. To show their impact on end users and apply distributed selection algorithm, we present an aggregation approach in this section.

4.1 Basic QoS Criteria

The *service related QoS criteria* have been discussed in several papers, e.g. response time, cost and availability. They belong to specific service and collected by QoS registry or clients report [8]. We assume there are *u* QoS parameters for service $s : Q^S \left(s\right) = \left\{q_1^s, q_2^s, ...q_u^s\right\}$, where $q_i^s (1 \le i \le u)$ is the ith QoS parameter.

The *link related QoS criteria* have been researched for many years in internet field but have not been substantially discussed in current service composition research. As mentioned above, they could have a great impact on the user experienced composition service QoS. We assume there are *v* QoS parameters for link $l : Q^L \left(l\right) = \left\{q_1^l, q_2^l, ...q_v^l\right\}$, where $q_i^l (1 \le i \le v)$ is the ith QoS parameter.

4.2 QoS Criteria Aggregation

The QoS criteria which belong to both services and links could be aggregated together. For example, the availability of a composition service is the multiplication of each service's availability and each link's availability. In fact, the aggregation between services and links could be expanded to many other application specific QoS criteria. Suppose the last x QoS parameters of link and service could be aggregated, the aggregated QoS of a link and a service is:

$$Q^A(l,s) = \left\{ q_1^s, \ldots\ldots q_{u-x}^s, q_1^l, \ldots\ldots q_{v-x}^l, q_1^a, \ldots\ldots q_x^a \right\} \tag{1}$$

$q_i^a = f_i(q_{v-x+i}^l, q_{u-x+i}^s)\left(1 \le i \le x, l \in Inlink(s)\right)$, $Inlink(s)$ is the set of links ended at service s, f_i is the function to aggregate the ith aggregative QoS parameter. By record $Q^A(l,s)$ to corresponding links, we can transform the service selection problem to link selection problem.

After the aggregation process, each link-service pair has $z = u + v - x$ QoS parameters. Thus a service path p has z QoS parameters too: $Q^P(p) = \left\{ q_1^p, q_2^p, \ldots\ldots q_n^p \right\}$. If QoS of a service path satisfy user's QoS constraints $Q^C = \left\{ q_1^c, q_2^c, \ldots\ldots q_z^c \right\}$, we call this path a *feasible service path.*

4.3 Scoring Service Path

Similar to [3], each service path has a *score* based on user interest:

$$Score(p) = \sum_{1 \le i \le z} w_i q_i^p, \left(0 \le w_i \le 1, \sum_{1 \le i \le z} w_i = 1 \right) \tag{2}$$

w_i is user's interest for the ith QoS parameter. We use (2) to evaluate a path and optimize selection algorithm, shows in section 5.

5 Distributed Service Selection Algorithm

We deduced the widely used Extended Bellman-Ford Algorithm (EBFA) in QoS routing to find feasible service path for critical task path. The basic idea of EBFA is: Find feasible path from source node (client) to nodes with n hops (services in task n), based on the result, iteratively find feasible path from source node to nodes with n+1 hops, until all candidate services has been computed.

A well-known problem of the EBFA is, in large network, each node may have to maintain a great number of feasible paths from source to it. For the specific problem of service selection, we propose a heuristic approach to effectively decrease the candidate path number and limit the traffic between QoS registries. It is a variation of limited path heuristic [2]. Briefly, our approach is: each service maintains at most K feasible paths which have the best scores. We call the algorithm *EBFA-K.*

A *distributed EBFA execution* process is: Based on previous tasks' selection result, a QoS registry computes a part of task sequence in critical task path and sends result to the next registry. The distributed execution illustrated in Fig.2. QoS registry 1 is responsible for find feasible service paths from client to task 1. It computes a set of candidate service path, sends result to registry 2. Registry 2 is responsible for selecting feasible service paths from task 1 to task 3. After computation, it sends result to registry 3. Registry 3 computes the final part of task path, send its result directly back to the first registry.

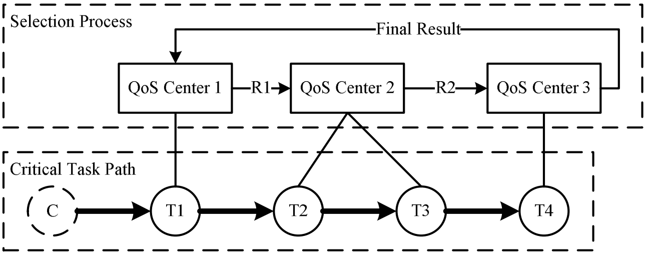


Fig. 2. Distributed execution process of EBFA

Let T denote the task sequence should be computed in current QoS registry. Let t' denote the last task computed in previous registry and $P' = Path\left(S\left(t'\right)\right)$ is the feasible path set from client to candidate services of task t' . s' represents the start service of link l . We assume the computation target is to minimize the score of service paths subject to user defined constraints. The EBFA-K algorithm on the ith QoS registry is illustrated in Fig.3.

```
EBFA-K$\left(P', T, Q^C\right)$
Aggregate all candidate service QoS to link QoS;
For each $t \in T$ in topological order
        For each $s \in S\left(t\right)$
                For each $l \in Inlink(s)$
                        RELAX-K$\left(Path(s'), s, l, Q^C\right)$
If all tasks computed
        Send $Path\left(S\left(t\right)\right)$ to QoS registry 1;
Else    Send $Path\left(S\left(t\right)\right)$ to QoS registry i+1;
Return;

RELAX$\left(Path(s'), s, l, Q^C\right)$
For each $p' \in Path(s')$
        If $Q^p\left(p'\right) + Q^a\left(l, s\right) < Q^c$
        Begin
                If $Q^p\left(p'\right) + Q^a\left(l, s\right) < Q^p\left(p\right), \forall p \in Path(s)$
                        Delete $p$;
                Else if $Q^p\left(p'\right) + Q^a\left(l, s\right) > Q^p\left(p\right), \forall p \in Path(s)$
                        Return;
                Insert $newpath = p' + l$ to $Path(s)$ ;
        End
        If $sizeof(Path(s)) > K$
                Remove the path with highest score in $Path(s)$ ;
Return;
```

Fig. 3. EBFA-K algorithm

Suppose there are n tasks in critical task path and each task has m candidate services, in the worst case, the time complexity of EBFA-K is $O\left(K^2nm^2\right)$, otherwise the worst time complexity of basic EBFA is $O(m^n)$ [2] and of Dijkstra algorithm in [6] is $O(m^2n^2)$. Whenever a task has been computed, we can delete all paths recorded in previous tasks, so that total space needed in EBFA-K is $O\left(Km\right)$. If all tasks belong to different QoS registries, traffic generated for computation between QoS registries is $O\left(Knm\right)$.

6 Experiments

We studied the effectiveness and performance of EBFA-K in different cases by a series of experiments. The experiments were conducted on several PCs with same configuration of a Pentium 4 1.6GHz CPU and 256MB RAM, running Debian LINUX. All hosts connected to a LAN through 100Mb/s Ethernet cards.

6.1 Evaluation Methodology

In experiment, we evaluate 3 cases: 1, centralized execution of EBFA; 2, centralized execution of EBFA-K; 3, decentralized execution of EBFA-K. In centralized execution of EBFA, service and link both have 2 QoS parameters, 1 of them could be aggregated. Task number in critical task path is set at 4 to 7 respectively. Service number in each task ranges from 2 to 10. In centralized execution of EBFA-K, we first compare its performance with EBFA, for its scalability, task number ranges from 10 to 40 with a step of 10 and service number in each task is from 5 to 50 with a step of 5. K is fixed at 5. We also evaluate the efficiency of EBFA-K for different QoS parameter number and different value of K. In this case, the task number is a constant value of 30, and service number in each task is 30. Total QoS parameter number is 10, 20 and 30, K is range from 1 to 10. In every experiment, we run 100 cases and each case have a set of random generated QoS parameters between 1 and 100.

An important problem of EBFA-K is it may discard feasible path or best path in execution. We use 2 criteria to study this feature:

1. *Success Ratio*: the times of finding out a feasible path in EBFA-K divided by the times of finding out a feasible path in EBFA, after executing the same set of cases. Shown in (3).

$$SuccessRatio = \frac{SuccessTime(EBFA-K)}{SuccessTime(EBFA)} \tag{3}$$

2. *Approximation*: For cases successfully executed in both EBFA and EBFA-K, sum up their result scores respectively, approximation is the sum of EBFA-K scores divided by the sum of EBFA scores. Shown in (4).

$$Approximation = \frac{SumofScores(EBFA-k)}{SumofScores(EBFA)} \tag{4}$$

In distributed execution of EBFA-K, we evaluate the network impact to total execution time. The distributed execution does not change the computation result. We test the EBFA-K in distributed execution when task number is 30 and service number in each task is 20 and 30 respectively, critical task path is randomly divided to hosts. The host number ranges from 2 to 6.

6.2 Result and Analysis

In Fig.4, we show that EBFA-K outperforms EBFA (shows in Fig.4(a)) very significantly. EBFA-K could scale up to 2000 of candidate services in total but EBFA could only execute in a scale of around 50 services. The execution time of EBFA-K is very competitive too. We do not show the execution time of EBFA-K in small scale for it is often under 10 milliseconds. This result is not a surprise because the limitation of path number kept in each service largely decreases the feasible path search time. Fig.5(a) shows the effect of K and QoS parameter number to execution time of EBFA-K. In our case, K is much less than the service number, so the execution time grows nearly linearly with K.

In experiment, EBFA-K achieves an excellent success ratio and approximation. In Fig.6(a), the success ratio is approximately 95% when K=4 and grows to 98% when K=10. Comparing to the great decrease of execution time, this success ratio is very acceptable. Interestingly, we found that in Fig.6(a), service number in each task has a positive effect to success ratio. This is because the more services in each task, the more paths would be kept. Fig.6(b) shows the approximation is about 99% when K>4.

Fig.5(b) shows that distributed execution time grows with the host number because of network delay between hosts. The distributed execution time would increase in a wide area network, but if we collected all service QoS information to 1 registry whenever we need a service composition, the time of transmitting would be unacceptable.

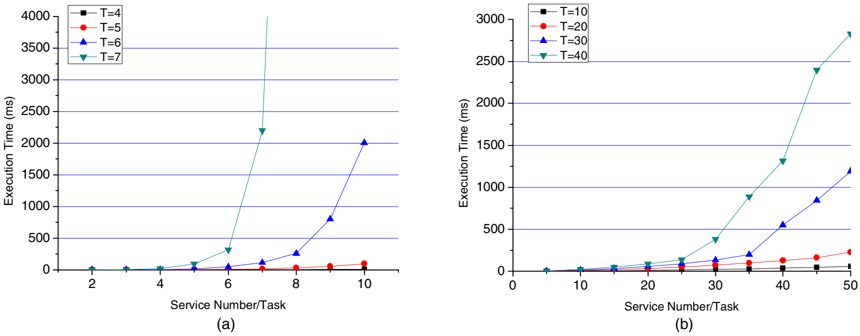


Fig. 4. (a) Execution time of EBFA; (b) Execution Time of EBFA-K

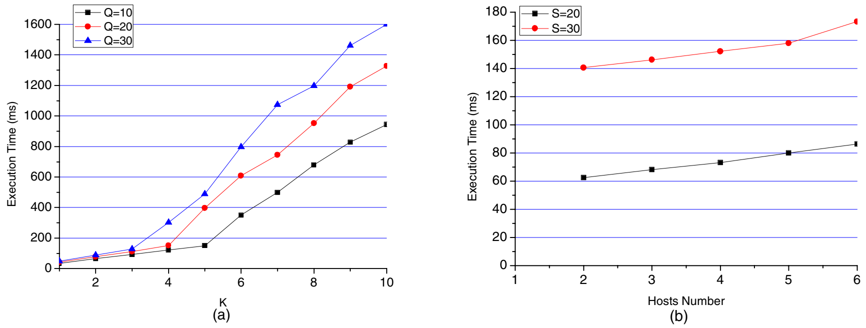


Fig. 5. (a) Execution time of EBFA-K in different K and QoS number; (b) Distributed execution time of EBFA-K

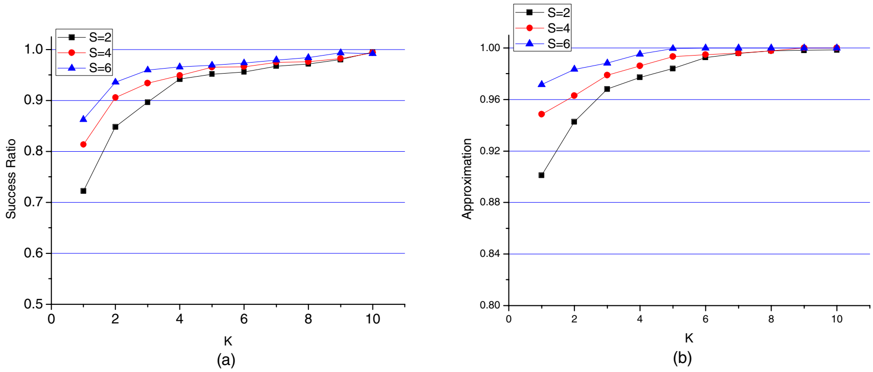


Fig. 6. (a) Success ratio of EBFA-K; (b) Approximation of EBFA-K

For space reasons, the following experiment results are not illustrated in figures. We also evaluated the limited path heuristic EBFA, called EBFA-LPK. Although the execution time of EBFA-LPK is better than our EBFA-K due to it does not need a sort operation, the approximation is much worse than EBFA-K. When K=10, EBFA-LPK achieved only 62% approximation and growing slowly with K. The execution time of Dijkstra algorithm is slightly faster than basic EBFA but much slower than EBFA-K.

7 Conclusion

In this paper, we present a distributed QoS registry architecture and a general QoS model for composition service. The distributed QoS registry could cooperate with other registries to enhance their capability. The model includes both network QoS and service QoS which could reliably reflect the user experienced QoS and convert service selection problem into path selection problem. To effectively solve the path selection problem, we modify basic EBFA algorithm in two aspects: Firstly, a decentralized execution approach, this modification breaks the assumption that one

QoS registry maintains all services; Secondly, a heuristic approach to limit the path number kept in services when executing EBFA. In experiment, we discuss the efficiency and accuracy of EBFA-K for different cases. In general, our algorithm shows good path selection result with excellent performance, which proved that our architecture would result in good user experience of QoS for composition service.

References

1. Zeng,L.Z.,Benatallah,B.,Dumas,M.,Kalagnanam,J.,ShengQuality,Q.Z.: Quality Driven Web Services Composition, In *Proceedings of the 12th International Conference on World Wide Web*, Budapest(2003), 411-421
2. Yuan,X.: On the Extended Bellman-Ford Algorithm to Solve Two-Constrained Quality of Service Routing Problems, In *Proceedings of the 8th International Conference on Computer Communications and Networks*(1999), 304 - 310
3. Zeng,L.Z.,Benatallah,B.,Ngu,A.H.H.,Dumas,M.,Kalagnanam,J.,Chang,H.: QoS-aware Middleware for Web Services Composition, *IEEE Transactions on Software Engineering, Vol.30(5)*, 2004, 311 - 327
4. Canfora,G.,Penta,M.D.,Esposito,R.,Villani,M.L.: An approach for QoS-aware Service Composition Based on Genetic Algorithms, In *Proceedings of the Genetic and evolutionary computation conference*, Washington DC(2005), 1069-1075
5. Yu,T. and Lin,K.J.: Service Selection Algorithms for Composing Complex Services with Multiple QoS Constraints, *In Proceedings of the 3th International Conference on Service Oriented Computing, LNCS 3826*, Springer-Verlag, 2005, 130-143
6. Gu,X.H.,Nahrstedt,K.,Chang,R.N.,Ward,C.: QoS-assured Service Composition in Managed Service Overlay Networks,In *Proceedings of the 23rd International Conference on Distributed Computing Systems*, 2003, 194-201
7. Gu,X.H.,Nahrstedt,K.,QoS-aware Service Composition for Large-scale Peer-to-peer Systems, *Grid Resource Management*, Kluwer Academic Publishers, 2004, 395-410
8. Liu,Y.T.,Ngu,A.H.,Zeng,L.Z.: QoS Computation and Policing in Dynamic Web Service Selection,In *Proceedings of the 13th International Conference on World Wide Web*,New York(2003),66-73
9. Day,J.,Deters,R.: Selecting the best web service, In *Proceedings of the Collaborative research Conference*, Markham,Ontario,Canada(2004), 293-307
10. Papazoglou,M.P.: Service-Oriented Computing: Concepts, Characteristics and Directions, In *Proceedings of the Fourth International Conference on Web Information Systems Engineering,* 2003, 3-12
11. Dustdar,S.,Schreiner W.: A Survey on Web services Composition, *International Journal of Web and Grid Services*, Vol.1(1),2005
12. Serhani, M.A., Dssouli, R., Hafid, A., Sahraoui, H.: A QoS Broker Based Architecture for Efficient Web Services Selection, In *Proceedings of International Conference on Web Services*, 2005. 113- 120,
13. Maximilien E. M., Singh, M. P.: Toward autonomic web services trust and selection, In *Proceedings of the Second International Conference on Service Oriented Computing*, 2004, 212-221.

RLinda: A Petri Net Based Implementation of the Linda Coordination Paradigm for Web Services Interactions[*]

J. Fabra, P. Álvarez, J.A. Bañares, and J. Ezpeleta

Instituto de Investigación en Ingeniería de Aragón (I3A)
Department of Computer Science and Systems Engineering, University of Zaragoza,
María de Luna 3, E-50018 Zaragoza, Spain
{jfabra, alvaper, banares, ezpeleta}@unizar.es

Abstract. The core functionality of Web-service middlewares tries to wrap existing business logics and make them accessible as Web services. Recently, well-known standardization initiatives have proposed some high-level declarative languages for the description of coordination protocols and the implementation of coordination middlewares. In parallel to these initiatives, an increasing interest on the use of classical coordination models on distributed environments has been shown. In this work we present a Linda-like coordination framework using *Petri nets*, which is executed by the *Renew* tool, a high-level Petri net interpreter developed in Java, and subsequently exposed as a Web service able to be used by other services for coordination purposes. The implementation is based on an extension of the original Linda model that improves the tuple representation capabilities and extends the matching functions used for the recovery of tuples from the coordination space. The efficiency of the proposed implementation has been empirically evaluated on a cluster computing environment, and its performances compared with the previously reported ones related to *JavaSpaces*.

Keywords: Service coordination, Linda, Petri nets, Tuple space benchmarks.

1 Introduction

The evolution of Web service middlewares requires the addition of some functionalities already present in traditional middlewares for enterprise application integration. Middlewares used in this context contain three main components [1,2]: 1) a workflow engine for the definition and execution of the business logic, 2) a message broker and 3) the necessary infrastructure to ensure the correctness and consistency of interactions (correct integration and execution of both, horizontal and business protocols).

[*] This work has been supported by the research project PIP086/2005, granted by the Diputación General de Aragón.

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 183–192, 2006.

The authors introduced in [3] a proposal of Web service middleware for the definition and execution of (complex) web processes which can interact using (complex) choreographies. The base formalism used in the proposal was the *Reference nets* paradigm [4], which is a subclass of the family of *Nets-within-Nets* [5]. As shown in the paper, the use of the same formalism for composition and coordination purposes helps in simplifying the development process. In the proposed framework, a coordination system based on the Linda paradigm was used as an intermediate language for the definition of the interactions (choreographies) among the involved Web services. This system acts as a message broker, allowing an abstract and technology-independent definition of the inter-services interactions. This point of view tries to bypass some vendors attempts to explicitly include their middleware system directly into WSDL specifications, violating the principle by which Web services must hide implementation details [6].

Tuple space-based coordination languages provide loosely coupled communication and coordination facilities regardless of any middleware programming model. Communication and coordination between processes is handled through a Tuple Space where processes put and withdraw tuples. This paradigm was conceived by Gelernter *et al.* [7] as part of the Linda coordination language. Its success in distributed systems has been due to a reduced set of basic operations, a data-driven coordination and a space and time uncoupled communication among processes that can cooperate without adapting or announcing themselves [8].

Different commercial organizations and research projects have concentrated on the development of Linda-based coordination systems in distributed environments. One of the most popular commercial implementations is JavaSpaces [9], provided by Sun Microsystems. Its specification inspired other Java implementations, such as GigaSpaces [10] and TSpaces [11] by IBM.

More recently, the popularity of XML as a mean for the coding and interchange of information in distributed environments made some XML-based tuple space implementations to appear, such as JXTASpaces [12] and XMLSpaces [13]. WorkSpaces [14] is a remarkable proposal that uses XMLSpaces for service composition.

As previously stated, the framework proposed in [3] was based on the use of the *Nets-within-Nets* paradigm, providing an environment for the definition and execution of Web services, implemented using the tool *Renew* [15], developed at the University of Hamburg. One of the components in the proposed framework was a Linda-like coordination system. This paper has two main purposes. The first one, to describe the implementation of that component. The second one, to measure some performance parameters and to compare them with JavaSpaces using the reference benchmark proposed in [16,17].

We are assuming the reader knows about Petri nets and Reference nets (see [18,4] for the main concepts used here).

The paper is organized as follows. Section 2 introduces a formal definition of tuple space and a set of matching functions. Section 3 describes an implementation of the Linda coordination system using the *Renew* tool. Section 4 presents the empirical performance measures obtained and the comparison with the ones reported in [17] for JavaSpaces. Finally, some conclusions are presented.

2 The Linda Coordination Model

Informally, Linda [7,19] is a coordination language which provides generative communication (communication by the production and consumption of passive data structures) over a shared space (a blackboard or a bag in a global associative memory). Data are represented as tuples (collections or sets of elements), so the shared space is called *tuple space*. Roughly speaking, a tuple space is a repository for a *multi-set* of tuples with, possibly, different lengths. In this paper nested tuples are allowed, being a nested tuple a kind of Lisp-like list, where elements can be either *atoms* or tuples (in our case, we are assuming that the empty tuple does not exist).

Let A be a set of typed elements, called *atoms*, for which a mapping $EQ_{\mathcal{A}} : \mathcal{A} \times \mathcal{A} \longrightarrow \{0,1\}$ is assumed to be defined. The set of tuples $\mathcal{T}_A$ is defined as follows:

1. for any $a \in A$, $[a]$ belongs to $\mathcal{T}_{\mathcal{A}}$
2. for any $[a] \in \mathcal{T}_A$, $[[a]]$ belongs to $\mathcal{T}_{\mathcal{A}}$
3. if $[e_1]$ and $[e_2]$ belong to $\mathcal{T}_{\mathcal{A}}$, then $[e_1\ e_2]$ also belongs to $\mathcal{T}_{\mathcal{A}}$

A tuple space is used by a set of processes in order to communicate and interact by means of the insertion and removal of tuples. A process wanting to insert a tuple t into a tuple space $\mathcal{T}_{\mathcal{A}}$ just needs to execute the $out_{\mathcal{A}}(t)$ operation. This is a non–blocking operation. On the other hand, the $in_{\mathcal{A}}$ Linda primitive can be used by a process in order to get a tuple from a subset of tuples of the tuple space. The set of tuples is usually indicated by means of a *pattern*. Let $\mathcal{T}_{\mathcal{A}}$ be a set of tuples. The *extended* tuple set, $\mathcal{T}_{\mathcal{A}^*}$, is defined as the tuple set $\mathcal{T}_{A\cup\{*\}}$. An element of the extended tuple set is called *pattern* (also *template* or *query template*).

A *matching function M* is a mapping $M : \mathcal{T}_{\mathcal{A}} \times \mathcal{T}_{\mathcal{A}^*} \longrightarrow \{0,1\}$. Given a tuple t and a pattern p defined over the same set, they are said to *match* in M if, and only if, $M(t,p) = 1$. The symbol $*$ introduced in the previous paragraph plays the role of a wildcard for the matching function.

Given a tuple space $\mathcal{T}_{\mathcal{A}}$, a matching function M, and a pattern p, the operation $t = in_{\mathcal{A}}(M,p)$ blocks the calling process until a M–matching tuple v in the tuple space is found such that $M(v,p) = 1$; v is then removed from the tuple space and assigned to t. The operation by which the tuple is found, removed, and assigned to v is an atomic operation. If more than one tuple exists M–matching with pattern v, one of the tuples is chosen in a non–deterministic way.

The third operation is a read operation. $t = rd_{\mathcal{A}}(M,p)$ is as $t = in_{\mathcal{A}}(M,p)$, with the only difference being that the chosen matching tuple is not removed from the tuple space.

By now, we are constraining ourselves to the case of four different matching functions, namely, *strong matching*, *weak matching*, *attribute matching* and *general attribute matching*.

The ***strong matching*** function, $SM_{\mathcal{A}}$, is defined as follows:

$$SM_{\mathcal{A}}: \quad \mathcal{T}_{\mathcal{A}} \times \mathcal{T}_{\mathcal{A}^*} \longrightarrow \{0,1\}$$

such that

$$[e_1 \dots e_n],[v_1 \dots v_m] \hookrightarrow (n=m) \wedge \bigwedge_{\alpha=1}^{n} \begin{cases} e_\alpha \in \mathcal{A} \wedge v_\alpha \in \mathcal{A} \ : \ EQ_{\mathcal{A}}(e_\alpha, v_\alpha) \\ e_\alpha \in \mathcal{A} \wedge v_\alpha = \ * \ : \ 1 \\ e_\alpha \in \mathcal{T}_{\mathcal{A}} \wedge v_\alpha \in \mathcal{T}_{\mathcal{A}^*} \ : \ SM_{\mathcal{A}}(e_\alpha, v_\alpha) \\ Else \ \ 0 \end{cases}$$

Notice that seeing a tuple as a representation of a tree, in the strong matching the template and the tuple must have exactly the same structure, and two leaves match if either both are "identical" (in the sense established by means of the $EQ_{\mathcal{A}}$ function) or the template leaf is a * (a wildcard). For example, the template $[a,[b,*],d]$ matches the tuple $[a,[b,c],d]$, but it does not match $[a,b,c,d]$.

The ***weak matching***, $WM_{\mathcal{A}}$, allows the correspondence between a tuple and a wildcard, and is defined as follows:

$$WM_{\mathcal{A}}: \quad \mathcal{T}_{\mathcal{A}} \times \mathcal{T}_{\mathcal{A}^*} \longrightarrow \{0,1\}$$

such that

$$[e_1 \dots e_n],[v_1 \dots v_m] \hookrightarrow (n=m) \wedge \bigwedge_{\alpha=1}^{n} \begin{cases} e_\alpha \in \mathcal{A} \wedge v_\alpha \in \mathcal{A} \ : \ EQ_{\mathcal{A}}(e_\alpha, v_\alpha) \\ e_\alpha \in \mathcal{T}_{\mathcal{A}} \cup \mathcal{A} \wedge v_\alpha = \ * \ : \ 1 \\ e_\alpha \in \mathcal{T}_{\mathcal{A}} \wedge v_\alpha \in \mathcal{T}_{\mathcal{A}^*} \ : \ WM_{\mathcal{A}}(e_\alpha, v_\alpha) \\ Else \ \ 0 \end{cases}$$

Notice that in this case a wildcard can match with both, an atom and a tuple. For example, the template $[a,*,d]$ matches the tuples $[a,[b,c],d]$ and $[a,b,d]$.

We have also considered two matching functions based on attribute matching and thought for XML scenarios. The ***attribute matching*** function allows to look for tuples matching a pattern `[attribute_name *]` at any level. The ***general attribute matching*** is the generalization of the previous one to a list of given attribute names.

3 A Petri Net-Based Implementation

Petri nets can help in providing an easy way of modeling concurrent and distributed systems, and also communication and coordination among processes. In this section we propose an implementation of a Linda coordination system in terms of Petri nets using the tool `Renew` [15]. This tool implements a graphical editor and an execution engine for the class of *Reference nets* [4], which is a subclass of the *Nets-within-Nets* family of Petri nets [5].

`Renew` is a high-level Petri net interpreter developed in Java, which allows an easy integration of reference nets and Java code associated to transitions (allowing to access Java code from the Petri net and also to access the Petri net from Java code). Renew implements a very efficient and flexible management of concurrent aspects, which suggests to consider it as a serious candidate for the implementation of a Linda coordination space. On the other hand, Renew uses a tuple concept that allows an easy and efficient implementation of the tuple concept used in Linda, since the inscription language of Renew includes *tuples* as a data type. We have extended this type with some basic methods to improve its functionality and to represent Linda-tuples as Renew-tuples, but without changing its original capabilities.

A client wanting to use our Linda implementation can access it by means of the invocation of operations `in`, `out` and `rd` via RMI or SOAP methods, passing an XML tuple description as parameter. For instance, the sentence $t_{xml} = in(p_{xml})$ will return in t_{xml} the XML description of a tuple matching the pattern described in p_{xml}. Figure 1 depicts the Renew implementation of the Linda server and shows the external interface, the client's point of view. Place `tupleSpace` is the tuple repository, where Linda tuples are stored as Renew tuples.

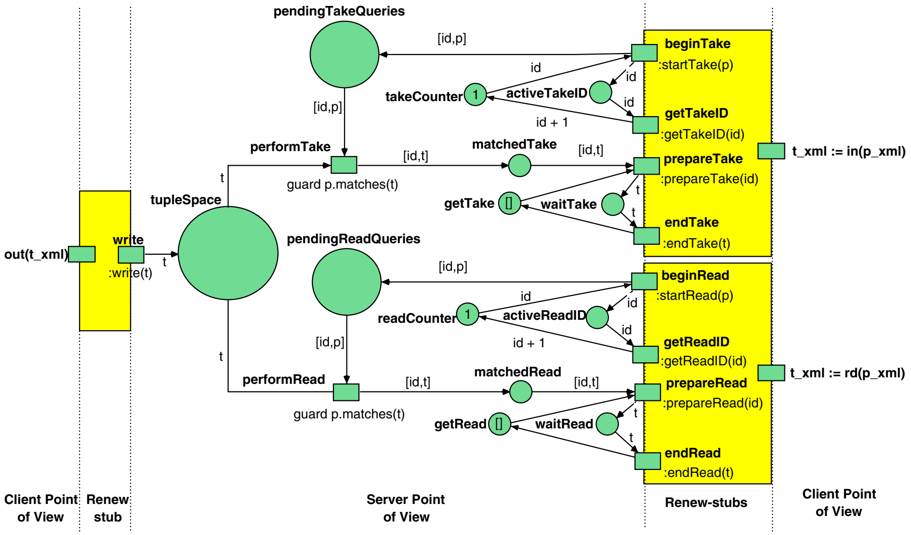


Fig. 1. The Linda coordination model

Let us now concentrate on the Petri net solution. From the client's point of view, `in`, `rd` and `out` are invoked as single operations. The `in` an `rd` operations are functions receiving a pattern as input parameter and returning a tuple, while `out` is a procedure with a tuple as unique input parameter. From the server point's of view, a *stub* is used to hide the sequence of operations required to perform the published operations. Renew manages *Renew-stubs*, a high-level interface description language which allows to encapsulate an ordered sequence of transition firings. To do that Renew provides *channels* as an unidirectional communication mechanism, allowing the transfer of parameters between the Petri net and the external generated stub-code:

```
int startTake(Tuple p) {  this:startTake(p); this:getTakeID(return);   }
Tuple endTake(int id)  {  this:prepareTake(id); this:endTake(return);  }
```

Let us trace the `in(p_xml)` operation. From a conceptual point of view, two steps are required: the one by which the pattern is inserted into the system, and the one by which a matching tuple is returned, when possible. Since Renew channels are unidirectional, each one of the previous steps is decomposed into two sub-steps, related by a system generated identifier: first, the pattern is inserted into the system (`this:startTake(t)`), returning an identifier (`this:getTakeID(return)`) associated to the pattern; second, the identifier is passed (`this:prepareTake(id)`) to obtain the corresponding tuple (`this:endTake(return)`).

Parsing from an XML description to the equivalent Renew tuple (from `p_xml` to `p`) and viceversa (from `t` to `t_xml`) is also performed in the Renew stub, being the following the code finally executed for the `in` operation:

```
XML_Tuple in(Tuple p_xml) {
  return parser.Tuple2XML(endTake(startTake(parser.XML2Tuple(p_xml))));
}
```

The `rd` operation is similar. The `out` operation is modeled with the input channel `:write(t)`, where `t` corresponds to the XML coded tuple. This way, the `out` operation results in a non-blocking operation, while `in` and `rd` result in blocking-operations since they involve the use of input and output synchronization channels.

Let us now concentrate on transition `performTake` in Figure 1, which applies the matching function. This transition has two input arcs: the arc from place `tupleSpace` that binds a matching tuple with variable `t`, and the arc from place `pendingTakeQueries`, which binds a pair `[requestID,pattern]` to variables `id` and `p`, respectively, and removes the matching tuple from the tuple space when the transition is fired. For this transition to be fired, the guard `p.matches(t)` must be open. We have implemented the matching functions described in Section 2. Pattern `p` carries the information of the matching function to be applied. The case of transition `performRead` is analogous, with the only difference being the type of arc related to the `tupleSpace` place: in this last case it is a `read` arc in Renew terminology, with the read semantics, which means that the token is not removed from the corresponding place.

4 Simulation, Evaluation and Results

Kapfhammer *et al.* have recently introduced in [17] a framework for tuple space benchmarking, collectively referred to as the Space bEnchmarking and TesTing moduLEs (SETTLE). They use the framework to measure the efficiency of

JavaSpaces [9] with the model reported in [16]. Let us now compare these results with the ones obtained by our implementation.

In the experiments, a set of Java clients access our Linda server. Every client performs a startup phase to define some local variables and set the benchmark parameters; then it iterates 1000 times the following cycle: execute an `out` operation, delay a random time ($T_{delay} \in [200{,}250]$ ms), and then retrieve the same tuple (operation `in`). When completed, the client shutdowns. A detailed justification of the used parameters can be found in [17]. To study the influence of the tuple's size we have used the same objects as in [17,20]: `NullEntry` objects (356 bytes), `StringEntry` objects (503 bytes), `DoubleArrEntry` objects (1031 bytes), and `FileEntry` objects (3493 bytes approximately). The matching function applied in [17] corresponds to the strong matching function, as defined in Section 2, where the $EQ_{\mathcal{A}}$ mapping is the `equals` function of the used object classes.

In the experiments the following measures have been taken: the throughput (mean number of `in`/`out` operations executed per second), and the response time (mean time between the instant in which an `in`/`out` operation arrives at the server and the time the result tuple is returned). Notice that both parameters are measured on the server's side to be independent of network delays. This way the results were obtained in a scenario similar to the one proposed in [17].

The benchmark was executed running q clients in q different nodes of a cluster using the Condor software for High Throughput Computing (HTC) over 22 nodes and an additional one dedicated to execute the Linda Server. The configuration for each node was a Genuine Intel Pentium 4 single processor workstation, 512 MB of RAM and a UDMA/133 SATA 7200 RPM disk subsystem running GNU/Linux with a customized 2.6.8-2 kernel compiled in order to use the `mwait`

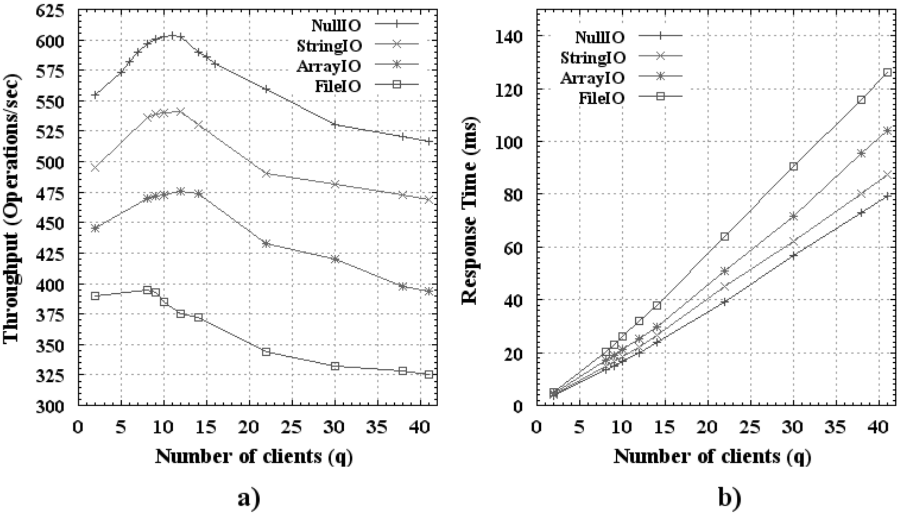


Fig. 2. Throughput and response time measurements for RLinda

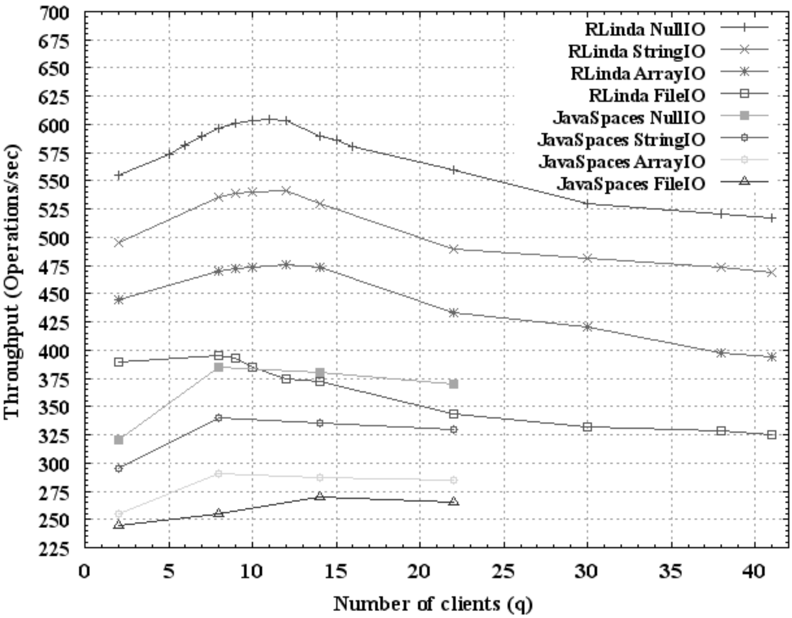


Fig. 3. Chart representing the measured throughput

instruction and an anticipatory input-output scheduler. The Java 1.4.2 compiler and virtual machine were run with the default configuration, and the Linda's server was allowed to get the maximum available physical memory.

Figure 2(a) provides a graphical representation of the throughput related to the number of concurrent clients q. The results shown that the heaviest operation is always an `out` or `in` operation using a `FileEntry` object. This type of object is the biggest one, which makes the matching process to require more CPU time for its processing. The best throughputs are always obtained when the number of clients q belongs to $[9, 14]$ (except when using the bigger `FileEntry` objects, which reaches the maximal values in the range $[2, 8]$). The existence of a certain performance decreasing threshold was previously reported by Zorman *et al.* in [16] for the JavaSpaces implementation. As stated in [17], "this threshold represents the number of concurrent clients that will cause tuple space throughput to *knee* as client response times continue to increase". With respect to the response time, Figure 2(b), it is clear that `NullEntry` tuples always take less time to execute than the others. For values beyond 41 concurrent clients the CPU execution time was so high that some connections were down, so it makes no sense to show results beyond this number.

Figure 3 compares the throughput obtained by JavaSpaces, as presented in [17] and the similar benchmarks executed with our implementation. Our implementation speeds up over a 75% in the best case. As reported in [16,17], performance significantly decreases in JavaSpaces when the number of concurrent clients exceeds a certain threshold around 8 concurrent clients. In our case, this threshold is around 12 clients. Notice that in [17] results were reported up to 22 clients, while our implementation allowed us to run up to 41 concurrent clients without degrading the system performances.

5 Conclusions

The framework for Web services definition and execution proposed in [3] used a Linda-like coordination system as an intermediate formalism for the definition and execution of complex interactions among Web services. Two main reasons made us think of a Petri net implementation for such coordination system, using the Renew tool. On the one hand, the fact that the rest of the framework is implemented in this formalism. On the other hand, the fact that Renew implements an efficient managing of concurrent aspects, which are necessary when implementing the set of Linda primitives.

In the paper we have described the proposed implementation, which has been compared, from a performance point of view, with the results for JavaSpaces reported in [17].

There are two main aspects to be developed in the future. The first one, the extension of the benchmark and experiments to more general classes of tuples and matching functions. The second one, to study a distributed version of our implementation.

Acknowledgements

The authors want to thank the three anonymous referees, whose remarks helped us to improve this paper.

References

1. G. Alonso, F. Casati, H. Kuno, and V. Machiraju: Web Services. Concepts, Architectures and Applications. Springer Verlag (2004)
2. R. Ten-Hove and P. Walker: Java Business Integration (JBI) 1.0, final release. Technical report, BEA Systems, IBM, Microsoft, SAP AG, and Siebel Systems (2005)
3. P. Álvarez, J. A. Bañares, and J. Ezpeleta: Approaching Web Service Coordination and Composition by Means of Petri Nets. The Case of the Nets-Within-Nets Paradigm. In: Third International Conference on Service Oriented Computing – ICSOC 2005. Amsterdam, The Netherlands, December 12-15, 2005. Volume 3826 of Lecture Notes in Computer Science. Springer Verlag (2005) 185–197
4. O. Kummer: Introduction to Petri Nets and Reference Nets. Sozionik Aktuell **1** (2001) 1–9
5. R. Valk: Petri Nets as Token Objects - An Introduction to Elementary Object Nets. In: 19th International Conference on Application and Theory of Petri Nets –ICATPN'98. Lisbon, Portugal, June 1998. Volume 1420 of Lecture Notes in Computer Science. Springer Verlag (1998) 1–25

6. S. Vinosky: Putting the "Web" into Web services. Web services interaction models. IEEE Internet Computing **6** (2002) 90–92
7. D. Gelernter: Generative communication in Linda. ACM Transactions on Programming Languages and Systems **7** (1985) 80–121
8. P. Álvarez, J. A. Bañares, and P.R. Muro-Medrano: An Architectural Pattern to Extend the Interaction Model between Web-Services: The Location-Based Service Context. In: First International Conference on Service Oriented Computing – ICSOC 2003. Trento, Italy, December 15-18, 2003. Volume 2910 of Lecture Notes in Computer Science. Springer Verlag (2003) 271–286
9. Sun Microsystems, Inc.: JavaSpaces Service Specification. Technical report, Sun Microsystems (2000)
10. GigaSpaces Technologies: GigaSpaces. Technical report, Sun Microsystems (2000)
11. AlphaWorks - TSpaces: `www.alphaworks.ibm.com/tech/tspaces`. (2003)
12. Project 'jxtaspaces': `http://jxtaspaces.jxta.org/`. (2004)
13. R. Tolksdorf and D. Glaubitz: Coordinating Web-Based Systems with Documents in XMLSpaces. In: 9th International Conference on Cooperative Information Systems. Trento, Italy, September 5-7, 2001. Volume 2172 of Lecture Notes in Computer Science. Springer Verlag (2001) 356–370
14. R. Tolksdorf: Workspaces: a Web-based Workflow Management System. IEEE Internet Computing **6** (2002) 18–26
15. O. Kummer, F. Wienberg, M. Duvigneau, J. Schumacher, M. Köhler, D. Moldt, H. Rölke, and R. Valk: An Extensible Editor and Simulation Engine for Petri Nets: Renew. In: 25th International Conference on Application and Theory of Petri Nets –ICATPN 2004. Bologna, Italy, June 2004. Volume 3099 of Lecture Notes in Computer Science. Springer Verlag (2004) 484–493
16. B. Zorman, G. M. Kapfhammer, and R. S. Roos: Creation and analysis of a JavaSpace-based genetic algorithm. In: Proceeedings of the 8th International Conference on Parallel and Distributed Processing Techniques and Applications –PDPTA '02. Las Vegas, Nevada, USA, June 24 - 27, 2002. Volume 3. CSREA Press (2002) 1107–1112
17. D. Fiedler, K. Walcott, T. Richardson, G. M. Kapfhammer, A. Amer, and P. K. Chrysanthis: Towards the Measurement of Tuple Space Performance. ACM SIGMETRICS Performance Evaluation Review **33** (2005) 51–62
18. T. Murata: Petri Nets: Properties, Analysis and Applications. In: Proceedings of the IEEE. (1989) 541–580 NewsletterInfo: Published as Proceedings of the IEEE, volume 77, number 4.
19. D. Gelernter: Multiple tuple spaces in Linda. In: Parallel Architectures and Languages Europe –PARLE '89. Eindhoven, The Netherlands, June 12-16, 1989. Volume 366 of Lecture Notes in Computer Science. Springer Verlag (1989) 20–27
20. M. S. Noble and S. Zlateva: Scientific computation with JavaSpaces. In: 9th International Conference on High-Performance Computing and Networking –HPCN Europe 2001. Amsterdam, The Netherlands, June 25-27, 2001. Volume 2110 of Lecture Notes in Computer Science. Springer Verlag (2001) 657–666

Making Informed Automated Trading a Reality

John Debenham and Simeon Simoff

Faculty of Information Technology, UTS, NSW, Australia
{debenham, simeon}@it.uts.edu.au
http://www.e-markets.org.au/

Abstract. Three core technologies are needed to fully automate the trading process: data mining, intelligent trading agents and virtual institutions in which informed trading agents can trade securely both with each other and with human agents in a natural way. This paper describes a demonstrable prototype e-trading system that integrates these three technologies and is available on the World Wide Web. This is part of a larger project that aims to make informed automated trading a reality.

1 Introduction

Trading involves the maintenance of effective business relationships, and is the complete process of: need identification, product brokering, supplier brokering, offer-exchange, contract negotiation, and contract execution. Three core technologies are needed to fully automate the trading process:

– data mining — real-time data mining technology to tap information flows from the marketplace and the World Wide Web, and to deliver timely information at the right granularity.
– trading agents — intelligent agents that are designed to operate in tandem with the real-time information flows received from the data mining systems.
– virtual institutions — virtual places on the World Wide Web in which informed trading agents can trade securely both with each other and with human agents in a natural way — not to be confused with the term "virtual organisations" as used in Grid computing.

This paper describes an e-trading system that integrates these three technologies. The e-Market Framework is available on the World Wide Web[1]. This project aims to make informed automated trading a reality, and develops further the "Curious Negotiator" framework [1]. This work does not address all of the issues in automated trading. For example, the work relies on developments in: XML and semantic web, secure data exchange, value chain management and financial services.

[1] http://e-markets.org.au

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 193–202, 2006.

2 Data Mining

We have designed information discovery and delivery agents that utilise text and network data mining for supporting real-time negotiation. This work has addressed the central issues of extracting relevant information from different online repositories with different formats, with possible duplicative and erroneous data. That is, we have addressed the central issues in extracting information from the World Wide Web. Our mining agents understand the influence that extracted information has on the subject of negotiation and takes that in account.

Real-time embedded data mining is an essential component of the proposed framework. In this framework the trading agents make their informed decisions, based on utilising two types of information: First, information extracted from the negotiation process (i.e. from the exchange of offers). Second, information from external sources, extracted and provided in condensed form.

The embedded data mining system provides the information extracted from the external sources. The system complements and services the information-based architecture developed in [2] and [3]. The data mining system initially constructs data sets that are "focused" on requested information. From the vast amount of information available in electronic form, we need to filter the information that is relevant to the information request. In our example, this will be the news, opinions, comments, white papers related to the five models of digital cameras. Technically, the automatic retrieval of the information pieces utilises the universal news bot architecture presented in [4]. Developed originally for news sites only, the approach is currently being extended to discussion boards and company white papers.

The "focused" data set is dynamically constructed in an iterative process. The data mining agent constructs the news data set according to the concepts in the query. Each concept is represented as a cluster of key terms (a term can include one or more words), defined by the proximity position of the frequent key terms. On each iteration the most frequent (terms) from the retrieved data set are extracted and considered to be related to the same concept. The extracted keywords are resubmitted to the search engine. The process of query submission, data retrieval and keyword extraction is repeated until the search results start to derail from the given topic.

The set of topics in the original request is used as a set of class labels. In our example we are interested in the evidence in support of each particular model camera model. A simple solution is for each model to introduce two labels — positive opinion and negative opinion, ending with ten labels. In the constructed "focused" data set, each news article is labelled with one of the values from this set of labels. An automated approach reported in [4] extends the tree-based approach proposed in [5].

Once the set is constructed, building the "advising model" is reduced to a classification data mining problem. As the model is communicated back to the information-based agent architecture, the classifier output should include all the possible class labels with an attached probability estimates for each class. Hence, we use probabilistic classifiers (e.g. Naïve Bayes, Bayesian Network classifiers [6]

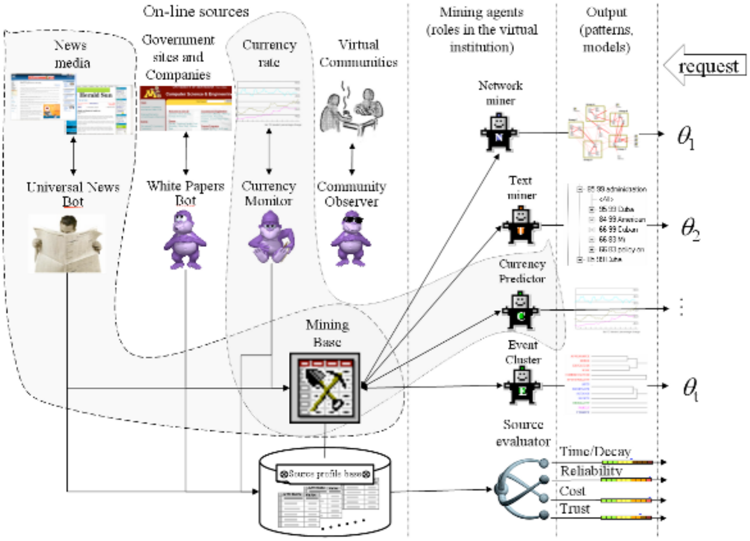


Fig. 1. The architecture of the agent-based data mining system

without the min-max selection of the class output [e.g., in a classifier based on Naïve Bayes algorithm, we calculate the posterior probability $\mathbb{P}_p(i)$ of each class $c(i)$ with respect to combinations of key terms and then return the tuples $< c(i), \mathbb{P}_p(i) >$ for all classes, not just the one with maximum $\mathbb{P}_p(i)$. In the case when we deal with range variables the data mining system returns the range within which is the estimated value. For example, the response to a request for an estimate of the rate of change between two currencies over specified period of time will be done in three steps: (i) the relative focused news data set will be updated for the specified period; (ii) the model that takes these news in account is updated, and; (iii) the output of the model is compared with requested ranges and the matching one is returned. The details of this part of the data mining system are presented in [7]. The currently used model is a modified linear model with an additional term that incorporates a news index Inews, which reflects the news effect on exchange rate. The current architecture of the data mining system in the e-market environment is shown in Figure 1. The $\{\theta_1, \ldots, \theta_t\}$ denote the output of the system to the information-based agent architecture. In addition, the data mining system provides parameters that define the "quality of the information", including:

- the time span of the "focused" data set, defined by the eldest and the latest information unit);
- estimates of the characteristics of the information sources, including reliability, trust and cost, that then are used by the information-based agent architecture.

3 Trading Agents

We have designed a new agent architecture founded on information theory. These "information-based" agents operate in real-time in response to market information flows. We have addressed the central issues of trust in the execution of contracts, and the reliability of information [3]. Our agents understand the value of building business relationships as a foundation for reliable trade. An inherent difficulty in automated trading — including e-procurement — is that it is generally multi-issue. Most of the work on multi-issue negotiation has focussed on one-to-one bargaining — for example [8]. There has been rather less interest in one-to-many, multi-issue auctions — [9] analyzes some possibilities. The main focus of our agents is their information and their strength of belief in its integrity. If their information is sufficiently certain then they may be able to estimate a utility function and to operate rationally in the accepted sense. However an agent may also be prepared to develop a semi-cooperative, non-utilitarian relationship with a trusted partner.

An agent called Π is the subject of this discussion. Π engages in multi-issue negotiation with a set of other agents: $\{\Omega_1, \cdots, \Omega_o\}$. The foundation for Π's operation is the information that is generated both by and because of its negotiation exchanges. Any message from one agent to another reveals information about the sender. Π also acquires information from the environment — including general information sources —to support its actions. Π uses ideas from information theory to process and summarize its information. Π's aim may not be "utility optimization" — it may not be aware of a utility function. If Π *does* know its utility function *and* if it aims to optimize its utility *then* Π may apply the principles of game theory to achieve its aim. The information-based approach does not to reject utility optimization — in general, the selection of a goal and strategy is secondary to the processing and summarizing of the information.

In addition to the information derived from its opponents, Π has access to a set of information sources $\{\Theta_1, \cdots, \Theta_t\}$ that may include the marketplace in which trading takes place, and general information sources such as news-feeds accessed via the Internet. Together, Π, $\{\Omega_1, \cdots, \Omega_o\}$ and $\{\Theta_1, \cdots, \Theta_t\}$ make up a multiagent system. The integrity of Π's information, including information extracted from the Internet, will decay in time. The way in which this decay occurs will depend on the type of information, and on the source from which it was drawn. Little appears to be known about how the integrity of real information, such as news-feeds, decays, although its validity can often be checked — "Is company X taking over company Y?" — by proactive action given a cooperative information source Θ_j. So Π has to consider how and when to refresh its decaying information.

Π triggers a goal, $g \in \mathcal{G}$, in two ways: first in response to a message received from an opponent $\{\Omega_i\}$ "I offer you €1 in exchange for an apple", and second in response to some need, $\nu \in \mathcal{N}$, "goodness, we've run out of coffee". In either case, Π is motivated by a need — either a need to strike a deal with a particular feature (such as acquiring coffee) or a general need to trade. Π's goals could be short-term such as obtaining some information "what is the time?", medium-term such as striking a deal with one of its opponents, or, rather longer-term such as building a (business) relationship with one of its opponents. So Π has a trigger mechanism T where: $T : \{\mathcal{X} \cup \mathcal{N}\} \to G$.

For each goal that Π commits to, it has a mechanism, G, for selecting a strategy to achieve it where $G : \mathcal{G} \times \mathcal{M} \to \mathcal{S}$ where $\mathcal{S}$ is the strategy library. A *strategy* s maps an information base into an action, $s(\mathcal{Y}^t) = z \in \mathcal{Z}$. Given a goal, g, and the current state of the social model m^t, a strategy: $s = G(g, m^t)$. Each strategy, s, consists of a *plan*, b_s and a *world model* (construction and revision) *function*, J_s, that constructs, and maintains the currency of, the strategy's *world model* W_s^t that consists of a set of probability distributions. A *plan* derives the agent's next action, z, on the basis of the agent's world model for that strategy and the current state of the social model: $z = b_s(W_s^t, m^t)$, and $z = s(\mathcal{Y}^t)$. J_s employs two forms of entropy-based inference:

- Maximum entropy inference, J_s^+, first constructs an *information base* $\mathcal{I}_s^t$ as a set of sentences expressed in $\mathcal{L}$ derived from $\mathcal{Y}^t$, and then from $\mathcal{I}_s^t$ constructs the world model, W_s^t, as a set of complete probability distributions.
- Given a prior world model, W_s^u, where $u < t$, minimum relative entropy inference, J_s^-, first constructs the incremental information base $\mathcal{I}_s^{(u,t)}$ of sentences derived from those in $\mathcal{Y}^t$ that were received between time u and time t, and then from W_s^u and $\mathcal{I}_s^{(u,t)}$ constructs a new world model, W_s^t.

The illocutions in the communication language $\mathcal{C}$ include information, [*info*]. The information received from general information sources will be expressed in terms defined by Π's ontology. The procedure for updating the world model as [*info*] is received follows. If at time u, Π receives a message containing [*info*] it is time-stamped and source-stamped $[\textit{info}]_{(\Omega,\Pi,u)}$, and placed in a repository $\mathcal{Y}^t$. If Π has an active plan, s, with model building function, J_s, then J_s is applied to $[\textit{info}]_{(\Omega,\Pi,u)}$ to derive constraints on some, or none, of Π's distributions. The extent to which those constraints are permitted to effect the distributions is determined by a value for the *reliability* of Ω, $R^t(\Pi, \Omega, O([\textit{info}]))$, where $O([\textit{info}])$ is the ontological context of [*info*].

In the absence of new [*info*] the integrity of distributions decays. If $D = (q_i)_{i=1}^n$ then we use a geometric model of decay:

$$q_i^{t+1} = (1 - \rho^D) \times d_i^D + \rho^D \times q_i^t, \text{ for } i = 1, \ldots, n \tag{1}$$

where $\rho^D \in (0, 1)$ is the decay rate. This raises the question of how to determine ρ^D. Just as an agent may know the decay limit distribution it may also know something about ρ^D. In the case of an information-overfed agent there is no harm in conservatively setting ρ^D "a bit on the low side" as the continually arriving [*info*] will sustain the estimate for D.

We now describe how new [*info*] is imported to the distributions. A single chunk of [*info*] may effect a number of distributions. Suppose that a chunk of [*info*] is received from Ω and that Π attaches the epistemic belief probability $R^t(\Pi, \Omega, O([\textit{info}]))$ to it. Each distribution models a facet of the world. Given a distribution $D^t = (q_i^t)_{i=1}^n$, q_i^t is the probability that the possible world ω_i for

D is the true world for D. The effect that a chunk $[\mathit{info}]$ has on distribution D is to enforce the set of linear constraints on D, $J^D_s([\mathit{info}])$. If the constraints $J^D_s([\mathit{info}])$ are taken by Π as valid then Π could update D to the posterior distribution $(p^{[\mathit{info}]}_i)^n_{i=1}$ that is the distribution with least relative entropy with respect to $(q^t_i)^n_{i=1}$ satisfying the constraint:

$$\sum_i \{p^{[\mathit{info}]}_i \;:\; J^D_s([\mathit{info}]) \text{ are all } \top \text{ in } \omega_i\} = 1. \tag{2}$$

But $R^t(\Pi, \Omega, O([\mathit{info}])) = r \in [0,1]$ and Π should only treat the $J^D_s([\mathit{info}])$ as valid if $r = 1$. In general r determines the extent to which the effect of $[\mathit{info}]$ on D is closer to $(p^{[\mathit{info}]}_i)^n_{i=1}$ or to the prior $(q^t_i)^n_{i=1}$ distribution by:

$$p^t_i = r \times p^{[\mathit{info}]}_i + (1 - r) \times q^t_i \tag{3}$$

But, we should only permit a new chunk of $[\mathit{info}]$ to influence D if doing so gives us new information. For example, if 5 minutes ago a trusted agent advises Π that the interest rate will go up by 1%, and 1 minute ago a very unreliable agent advises Π that the interest rate may go up by 0.5%, then the second unreliable chunk should not be permitted to 'overwrite' the first.

Information Reliability. We estimate $R^t(\Pi, \Omega, O([\mathit{info}]))$ by measuring the error in information. Π's plans will have constructed a set of distributions. We measure the 'error' in information as the error in the effect that information has on each of Π's distributions. Suppose that a chunk of $[\mathit{info}]$ is received from agent Ω at time s and is verified at some later time t. For example, a chunk of information could be "the interest rate will rise by 0.5% next week", and suppose that the interest rate actually rises by 0.25% — call that correct information $[\mathit{fact}]$. What does all this tell agent Π about agent Ω's reliability? Consider one of Π's distributions D that is $\{q^s_i\}$ at time s. Let $(p^{[\mathit{info}]}_i)^n_{i=1}$ be the minimum relative entropy distribution given that $[\mathit{info}]$ has been received as calculated in Eqn. 2, and let $(p^{[\mathit{fact}]}_i)^n_{i=1}$ be that distribution if $[\mathit{fact}]$ had been received instead. Suppose that the reliability estimate for distribution D was R^s_D. This section is concerned with what R^s_D should have been in the light of knowing *now*, at time t, that $[\mathit{info}]$ should have been $[\mathit{fact}]$, and how that knowledge effects our current reliability estimate for D, $R^t(\Pi, \Omega, O([\mathit{info}]))$.

The idea of Eqn. 3, is that the current value of r should be such that, *on average*, $(p^s_i)^n_{i=1}$ will be seen to be "close to" $(p^{[\mathit{fact}]}_i)^n_{i=1}$ when we eventually discover $[\mathit{fact}]$ — no matter whether or not $[\mathit{info}]$ was used to update D. That is, given $[\mathit{info}]$, $[\mathit{fact}]$ and the prior $(q^s_i)^n_{i=1}$, calculate $(p^{[\mathit{info}]}_i)^n_{i=1}$ and $(p^{[\mathit{fact}]}_i)^n_{i=1}$ using Eqn. 2. Then the *observed reliability* for distribution D, $R^{([\mathit{info}]|[\mathit{fact}])}_D$, on the basis of the verification of $[\mathit{info}]$ with $[\mathit{fact}]$ is the value of r that minimises the Kullback-Leibler distance between $(p^s_i)^n_{i=1}$ and $(p^{[\mathit{fact}]}_i)^n_{i=1}$:

$$\arg\min_r \sum_{i=1}^n (r \cdot p^{[\mathit{info}]}_i + (1-r) \cdot q^s_i) \log \frac{r \cdot p^{[\mathit{info}]}_i + (1-r) \cdot q^s_i}{p^{[\mathit{fact}]}_i}$$

If $E^{[\mathit{info}]}$ is the set of distributions that [*info*] effects, then the overall *observed reliability* on the basis of the verification of [*info*] with [*fact*] is: $R^{([\mathit{info}]|[\mathit{fact}])} = 1-(\max_{D\in E^{[\mathit{info}]}} |1-R_D^{([\mathit{info}]|[\mathit{fact}])}|)$. Then for each ontological context o_j, at time t when, perhaps, a chunk of [*info*], with $O([\mathit{info}]) = o_k$, may have been verified with [*fact*]:

$$R^{t+1}(\Pi, \Omega, o_j) = (1-\rho) \times R^t(\Pi, \Omega, o_j) + \rho \times R^{([\mathit{info}]|[\mathit{fact}])} \times \mathrm{Sem}(o_j, o_k) \quad (4)$$

where $\mathrm{Sem}(\cdot,\cdot) : O \times O \to [0,1]$ measures the semantic distance between two sections of the ontology, and ρ is the learning rate. Over time, Π notes the ontological context of the various chunks of [*info*] received from Ω and over the various ontological contexts calculates the relative frequency, $P^t(o_j)$, of these contexts, $o_j = O([\mathit{info}])$. This leads to a overall expectation of the *reliability* that agent Π has for agent Ω:

$$R^t(\Pi, \Omega) = \sum_j P^t(o_j) \times R^t(\Pi, \Omega, o_j)$$

4 Virtual Institutions

This work is done on collaboration with the Spanish Governments IIIA Laboratory in Barcelona. Electronic Institutions are software systems composed of autonomous agents, that interact according to predefined conventions on language and protocol and that guarantee that certain norms of behaviour are enforced. Virtual Institutions enable rich interaction, based on natural language and embodiment of humans and software agents in a “liveable” vibrant environment. This view permits agents to behave autonomously and take their decisions freely up to the limits imposed by the set of *norms* of the institution. An important consequence of embedding agents in a virtual institution is that the predefined conventions on language and protocol greatly simplify the design of the agents. A Virtual Institution is in a sense a natural extension of the social concept of institutions as regulatory systems that shape human interactions [10].

Virtual Institutions are electronic environments designed to meet the following requirements towards their inhabitants:

- enable institutional commitments including structured language and norms of behaviour which enable reliable interaction between autonomous agents and between human and autonomous agents;
- enable rich interaction, based on natural language and embodiment of humans and software agents in a “liveable” vibrant environment.

The first requirement has been addressed to some extent by the Electronic Institutions (EI) methodology and technology for multi-agent systems, developed in the Spanish Government’s IIIA Laboratory in Barcelona [10]. The EI environment is oriented towards the engineering of multiagent systems. The Electronic Institution is an environment populated by autonomous software agents that interact according to predefined conventions on language and protocol. Following the metaphor of social institutions, Electronic Institutions guarantee that certain norms of behaviour are enforced. This view permits that agents behave autonomously and make their decisions freely up to the limits imposed by the set of norms of the institution. The interaction in such environment is regulated for software agents. The human, however, is "excluded" from the electronic institution.

The second requirement is supported to some extent by the distributed 3D Virtual Worlds technology. Emulating and extending the physical world in which we live, Virtual Worlds offer rich environment for a variety of human activities and multi-mode interaction. Both humans and software agents are embedded and visualised in such 3D environments as avatars, through which they communicate. The inhabitants of virtual worlds are aware of where they are and who is there — elements of the presence that are excluded from the current paradigm of e-Commerce environments. Following the metaphor of the physical world, these environments do not impose any regulations (in terms of language) on the interactions and any restrictions (in terms of norms of behaviour). When this encourages the social aspect of interactions and establishment of networks, these environments do not provide means for enabling some behavioural norms, for example, fulfilling commitments, penalisation for misbehaviour and others.

Virtual Institutions addressed both requirements, retaining the features and advantages of the above discussed approaches. They can be seen as the logical evolution and merger of the two streams of development of environments that can host electronic markets as mixed societies of humans and software agents.

Technologically, Virtual Institutions are implemented following a three-layered framework, which provides deep integration of Electronic Institution technology and Virtual Worlds technology [11]. The framework is illustrated in Figure 2. The Electronic Institution Layer hosts the environments that support the Electronic Institutions technological component: the graphical EI specification designer ISLANDER and the runtime component AMELI [12]. At runtime, the Electronic Institution layer loads the institution specification and mediates agents interactions while enforcing institutional rules and norms.

The Communication Layer connects causally the Electronic Institutions layer with the 3D representation of the institution, which resides in the Social layer. The causal connection is the integrator. It enables the Electronic Institution layer to respond to changes in the 3D representation (for example, to respond to the human activities there), and passes back the response of the Electronic Institution layer in order to modify the corresponding 3D environment and maintain the consistency of the Virtual Institution. Virtual Institution representation is a graph and its topology can structure the space of the virtual environment in different ways. This is the responsibility of the Social layer. In this implementation the layer is represented in terms of a 3D Virtual World technology, structured around rooms, avatars, doors (for transitions) and other graphical elements. Technically, the Social layer is currently utilising Adobe Atmosphere

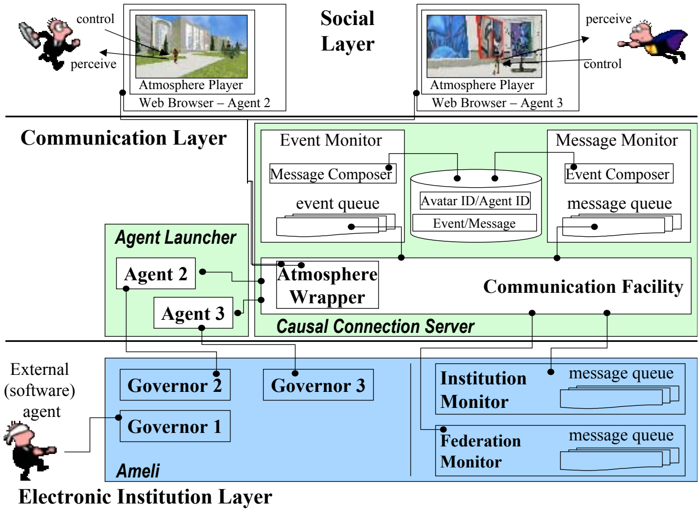


Fig. 2. The three layer architecture and its implementation

virtual world technology. The design of the 3D World of the Virtual Institution is developed with the Annotation Editor, which ideally should take as an input a specification of the Electronic Institution layer and produce an initial layout of the 3D space. Currently, part of the work is done manually by a designer.

The core technology — the Causal Connection Server, enables the Communication Layer to act in two directions. Technically, in direction from the Electronic Institution layer, messages uttered by an agent have immediate impact in the Social layer. Transition of the agent between scenes in the Electronic Institution layer, for example, must let the corresponding avatar move within the Virtual World space accordingly. In the other direction, events caused by the actions of the human avatar in the Virtual World are transferred to the Electronic Institution layer and passed to an agent. This implies that actions forbidden to the agent by the norms of the institution (encoded in the Electronic Institution layer), cannot be performed by the human. For example, if a human needs to register first before leaving for the auction space, the corresponding agent is not allowed to leave the registration scene. Consequently, the avatar is not permitted to open the corresponding door to the auction (see [11] for technical details of the implementation of the Causal Connection Server).

Virtual Institutions are immersive environments and as such go beyond the catalogue-style markets with form-based interaction approaches currently dominating the World Wide Web. Embedding traders (whether humans or software agents) as avatars in the electronic market space on the Web positions them literally “in” the World Wide Web rather than “on” it.

5 Conclusions

A demonstrable prototype e-Market system permits both human and software agents to trade with each other on the World Wide Web. The main contributions described are: the broadly-based and “focussed” data mining systems, the intelligent agent architecture founded on information theory, and the abstract synthesis of the virtual worlds and the electronic institutions paradigms to form “virtual institutions”. These three technologies combine to present our vision of the World Wide Web marketplaces of tomorrow.

References

1. Simoff, S., Debenham, J.: Curious negotiator. In M. Klusch, S.O., Shehory, O., eds.: proceedings 6th International Workshop Cooperative Information Agents VI CIA2002, Madrid, Spain, Springer-Verlag: Heidelberg, Germany (2002) 104–111
2. Debenham, J.: Bargaining with information. In Jennings, N., Sierra, C., Sonenberg, L., Tambe, M., eds.: Proceedings Third International Conference on Autonomous Agents and Multi Agent Systems AAMAS-2004, ACM (2004) 664 – 671
3. Sierra, C., Debenham, J.: An information-based model for trust. In Dignum, F., Dignum, V., Koenig, S., Kraus, S., Singh, M., Wooldridge, M., eds.: Proceedings Fourth International Conference on Autonomous Agents and Multi Agent Systems AAMAS-2005, Utrecht, The Netherlands, ACM Press, New York (2005) 497 – 504
4. Zhang, D., Simoff, S.: Informing the Curious Negotiator: Automatic news extraction from the Internet. In: Proceedings 3’rd Australasian Data Mining Conference, Cairns, Australia (2004) 55–72
5. Reis, D., Golgher, P.B., Silva, A., Laender, A.: Automatic web news extraction using tree edit distance. In: Proceedings of the 13’th International Conference on the World Wide Web, New York (2004) 502–511
6. Ramoni, M., Sebastiani, P.: Bayesian methods. In: Intelligent Data Analysis. Springer-Verlag: Heidelberg, Germany (2003) 132–168
7. Zhang, D., Simoff, S., Debenham, J.: Exchange rate modelling using news articles and economic data. In: Proceedings of The 18th Australian Joint Conference on Artificial Intelligence, Sydney, Australia, Springer-Verlag: Heidelberg, Germany (2005)
8. Faratin, P., Sierra, C., Jennings, N.: Using similarity criteria to make issue tradeoffs in automated negotiation. Journal of Artificial Intelligence **142** (2003) 205–237
9. Debenham, J.: Auctions and bidding with information. In Faratin, P., Rodriguez-Aguilar, J., eds.: Proceedings Agent-Mediated Electronic Commerce VI: AMEC. (2004) 15 – 28
10. Arcos, J.L., Esteva, M., Noriega, P., Rodríguez, J.A., Sierra, C.: Environment engineering for multiagent systems. Journal on Engineering Applications of Artificial Intelligence **18** (2005)
11. Bogdanovych, A., Berger, H., Simoff, S., Sierra, C.: Narrowing the gap between humans and agents in e-commerce: 3D electronic institutions. In Bauknecht, K., Pröll, B., Werthner, H., eds.: E-Commerce and Web Technologies, Proceedings of the 6th International Conference, EC-Web 2005, Copenhagen, Denmark, Springer-Verlag: Heidelberg, Germany (2005) 128–137
12. (Electronic institution development environment: http://e-institutor.iiia.csic.es/)

An Analysis of Service Trading Architectures

Manuel Resinas, Pablo Fernandez, and Rafael Corchuelo

ETS Ingenieria Informatica
Universidad de Sevilla, Spain
http://www.tdg-seville.info

Abstract. Automating the creation and management of SLAs in electronic commerce scenarios brings many advantages, such as increasing the speed in the contracting process or allowing providers to deploy an automated provision of services based on those SLAs. We focus on the service trading process, which is the process of locating, selecting, negotiating, and creating SLAs. This process can be applied to a variety of scenarios and, hence, their requirements are also very different. Despite some service trading architectures have been proposed, currently there is no analysis about which one fits better in each scenario. In this paper, we define a set of properties for abstract service trading architectures based on an analysis of several practical scenarios. Then, we use it to analyse and compare the most relevant abstract architectures for service trading. In so doing, the main contribution of this article is a first approach to settle the basis for a qualitative selection of the best architecture for similar trading scenarios.

1 Introduction

Service level agreements (SLAs) are used by many different service industries to grant guarantees about how a service will be provided or consumed by establishing both functional and non-functional requirements that must be fulfilled by both parties during the service development. The application of Internet-based technologies for electronic commerce to establish and manage these SLAs offers significant advantages to the traditional use of SLAs [1]. Specifically, automating the creation and management of SLAs, so that the human participation in the process is reduced to the minimum, brings benefits such as cutting down the cost of reaching an agreement, increasing the speed in the contracting process and allowing providers to deploy an automated provision of services based on the SLAs agreed with their customers [2].

We define *service trading process* as the process of locating, selecting, negotiating, and creating SLAs. Therefore, the service trading process is a subprocess that covers the information and negotiation phases of the more general contracting process [3]. The characteristics of the service trading process depend on the particular scenario where it is developed. These scenarios are very diverse and can range from a traditional supply chain to dynamically selecting the best VoIP (Voice over IP) provider or contracting or renegotiating a contract with an ISP

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 203–212, 2006.

(see Section 2 for more information). As the scenarios are very different, the requirements for each of them are also diverse. Therefore, we argue that there is no one unique solution for service trading but we must choose the most appropriate option for each situation.

We focus on abstract architectures for service trading, which are specifications that define a set of elements for service trading. These abstract architectures can be later implemented by using different technologies and applied for different problem domains. Our goal in this paper is to define a set of properties for these abstract service trading architectures based on an analysis of several service trading scenarios. These properties enable the analysis and comparison of those abstract architectures, which is a necessary step to select the most appropriate architecture in each scenario.

The paper is structured as follows. First, in Section 2, we present four service trading scenarios that serve as the basis for the set of properties for abstract service trading architectures presented in Section 3. In Section 4, these properties are used to analyse and compare the most relevant abstract architectures for service trading. Finally, we conclude in Section 5.

2 Scenarios

In this section, we describe a set of scenarios that correspond to different cases of service trading that have been selected based on the diversity of features, stakeholders and domains.

Scenario 1. Service consumer looking for ISPs. This case relate to a mid-large size company that looks for an ISP (Internet Service Provider). In this scenario, a company publishes its demands and wait for the ISPs to make their offers. In so doing, the company has a passive role while the providers act as active organisations searching for customers. In this domain and from the point of view of the company, it is appealing to have a periodic renegotiation of the service. Furthermore, a high level of automation in the service trading enables that every renegotiation is open to different ISPs in order to select the best possible in each case; in this way, it is boosted a dynamic market where each provider look forward competitive offers adjusted at their capabilities in each moment. An additional issue is the strict temporal sequencing of the service trading process. The trading process should coherently encompass the stages to fit the temporal constraints for the company to avoid problems such as a lack of the service due to the change of ISP.

In this scenario, the QoS terms during the SLA establishment are a key factor. In this way, an interesting feature is to be able to automatically negotiate such features. Concerning the decision-making process, the information known about providers is the most important element; i.e. the reputation of provider or the historic knowledge based on previous trading process. Lastly, participant organisations should have the guarantee that the agreements reached by the system are legitimate.

Scenario 2. Computing services provider. In this case, a company offers computing services to other organisations. In particular, this case is becoming increasingly popular in research fields with intensive computational needs such as bioinformatics. In a concrete manner, the company in this scenario can be described as a computing service provider that receives demands from other organisations in terms of computing jobs to be developed.

In this scenario it should be allowed for the company to specify offers that optimise the usage of its resources. Specifically, in a computing company, unused (or low used) resources means a decrease in the recovery of the initial investment. In so doing, offers should vary based on the resource usage and the set of SLAs the company has reached with its customers. Closely related with those ideas, from the perspective of the customer, a negotiation of the terms of the SLA is an interesting issue to be addressed. Moreover, this negotiation process can be used by the provider to slightly adapt the final SLA and make concessions or restrictions in order to optimise the current usage level of its resources.

Additionally, as those offers are tightly adjusted to the resource status in each moment, the decision making infrastructure should also take into account this information as a first level element before establishing new commitments with a customer. Finally, it is interesting to remark that, unlike the previous case, the reputation information is not an important issue from the perspective of the computing provider.

Scenario 3. Company delegating to a trader specialized in VoIP. This case describes a company that delegates its telephony needs to a trader that handle its requests and locate the best possible VoIP(Telephony through Internet) provider for each call. This trader represents an organisation that makes profit acting as a facilitator between end-user companies and VoIP providers. These providers offers different services characteristics (e.g. guaranteed bandwidth) or restrictions (e.g. Some of them could only operate between certain countries). In so doing, this trader offers a service of calls management by creating concrete agreements for each call (or set of calls) with the telephony provider taking into account the preferences of the company: e.g. minimising cost.

The information about the telephony service providers can be divided into two sets: First, a set of information describing the capabilities of the provider in general terms such as the operational countries or time slots classification (*Peak hours*, *reduced cost hours*, etc.); this set can be used to create a selection of potential providers. Second, in each moment, when handling an specific call, the trader can ask about *last minute* offers from the providers; this offers would be based on the resource status of provider (as in the previous scenario). In so doing, based on the information harvested, the trader can construct the specific SLA proposal the most appealing provider and, finally, if it agrees, the final SLA is established and the call can be carried out.

Scenario 4. A generic service trader in a supply-chain. One of the scenarios where service trading fits better is supply-chains, where each organisation create added value by composing services from different providers. In this case, a trader of services represents organisations that create high-level services based on the composition of lower-level services. Many examples can be found in the literature from telecommunications domain [4] to the transport domain [5]. This idea of supply-chain can be isolated from a specific domain and, hence, the elements and requirements expressed in this scenario are mostly valid for the majority of supply-chains independently of the nature of the services supplied.

In this scenario, the key point to be addressed is an efficient composition of services that adds value to the customer whose the trader sells its services. To achieve its goals, an aspect to be addressed is the adaptability to different markets. To a service trader the ability to understand different ontologies is important because it allows him to communicate with different markets or providers. Closely related with the previous ideas, information harvested about different providers is necessary for an efficient decision making process of selection of services. For each potential service provider, the trader should ask to several sources: the provider itself for information about the capabilities it claims to have, and third parties that can also supply important information about the reputation of a certain provider.

In order to construct a composed service, the trader should agree several SLAs with providers for each of the services that will be composed. In this way, during the establishment of every SLA, three processes are relevant: (i) a classification of the proposals (coming from the providers); (ii) a selection of the most promising proposals; (iii) a decision about the handling process for each of the selected proposals: e.g. whether we negotiate them or not.

3 Properties of Service Trading Architectures

In this section, we present a set of properties describing features of abstract architectures for service trading. These properties are derived from the four scenarios described in Section 2. For each property a reference to the related scenario is supplied; concretely, this references are based in the special importance of the property for the specified scenario.

1. *External discovery (S.1, S.2, S.3 and S.4)*: We say an abstract architecture has an external discovery process if it uses an external infrastructure (e.g. an external registry) to obtain the list of potential parties that demand (or supply) a service that other party provides (or needs). Alternatively, the process is internal if no external infrastructure is used, for example, if the list of potential parties is directly provided by the user.
2. *Knowledge adaptation (S.4)*: An abstract architecture has knowledge adaptation [6] [7] if it provides elements to adapt the local knowledge model to the appropriate discovery infrastructure, making independent the characteristics of the market modelled by the discovery service to the rest of architecture.
3. *Market observation(S.3 and S.4)*: An abstract architecture with market observation monitors the changes in the market through observation of the information provided by external discovery infrastructures and informs about these changes to the elements of the architecture.

4. *Symmetric architecture for providers and consumers (S.4)*: An abstract architecture is symmetric if both service provider and consumer can start the service trading process and there is no commitment as to which party advertises and which party queries to the discovery service. Alternatively, an abstract architecture is asymmetric if only one consumer or provider can start the service trading process.
5. *Information query (S.3 and S.4)*: An information query is an inquiry made by one party to another to obtain more detailed information about it or about the service it provides or demands. Therefore, for an abstract architecture to support information queries, it must have mechanisms to query services or to respond to those queries.
6. *World model (S.1, S.3 and S.4)*: An abstract architecture builds a world model if it analyses previous interactions with the elements external to the architecture, such as other parties or the discovery services, and uses the results to make better decisions [8] during the service trading process.
7. *Third party information (S.1, S.3 and S.4)*: This property represent that a third party is explicitly queried to obtain information related to another. For instance, to obtain information about its reputation or its geographical location. In this case, a protocol to carry out this query as well as a shared taxonomy of terms must be supported by the architecture.
8. *Information managed about the parties (S.1, S.2, S.3 and S.4)*: There are three types of information that can be managed about the parties: *service information*, that is, information about the functional and non-functional characteristics of the service; *trading information* or information about the features of the trading process followed by the party; and *party information*, i.e. information about the party that provides or demands a service, such as its reputation or its geographical situation.
9. *Proposals preselection (S.3 and S.4)*: An abstract architecture has a proposals preselection process if, before starting an agreement creation process or a negotiation, it ranks and/or filters the proposals that it has received or built based on criteria previously specified.
10. *Agreement creation mechanisms (S.1, S.2 and S.4)*: An abstract architecture has multiple agreement creation mechanisms if it supports different protocols to reach to an agreement. These mechanisms can range from a take-it-or-leave-it protocol [9] to a bilateral negotiation or an auction protocol [8].
11. *Notary (S.1 and S.4)*: An abstract architecture has this property if it provides any mechanism to guarantee that the agreement created between the two parties is reliable and non-repudiable. We say an agreement is reliable if both parties are signing and accepting the same previously agreed document.
12. *Decommitment from previously established agreements (S.1 and S.4)*: An abstract architecture supports the decommitment [10] from previously established agreements if it can revoke previous agreements before the execution of the service, possibly by paying some compensation to the other party. This implies the implementation of any decommit protocol and the mechanisms to decide when a decommit is profitable for it.

13. *Capacity estimator (S.2 and S.3)*: An abstract architecture may make use of a capacity estimator to determine whether the provider can provision a certain agreement enabling a finer control about its resources and the implications of the agreements created [2].
14. *Trading protocols (S.1, S.2, S.3 and S.4)*: A trading protocol is a set of stages (e.g. *advertisement, proposal submission, negotiation, resolution, etcetera.*) cross-linked in accordance to some temporal constraints and bounded to some choreographies. The temporal restrictions specify a set of constraints about the life-cycle of the trading process. These restrictions can vary from simple fixed temporal points (e.g. *End by 14:00 of 14th, March*) to complex relationships amongst the durations of some stages (e.g. *Information stage starts in the middle of the discovery stage*). Therefore, an abstract architecture that supports different trading protocols must be able to deal with different temporal constraints on the stages of the service trading process.
15. *Creation of agreements for composed services (S.4)*: A composed service [11] is a service whose implementation is based on the execution of other services that may be provided by external entities and, hence, there may exist agreements regulating that execution. The support for creating agreements for composed services can vary significantly, from simple dependencies between the services such as "*I want either an agreement on all different services or no agreement at all*" to taking into account the service level properties desired for the composed service.
16. *Cooperative or non-cooperative agreement creation (S.1, S.2, S.3 and S.4)*: An abstract architecture supports non-cooperative agreement creation when it acts as a self-interested party reaching agreements with other self-interested parties. Alternatively, an abstract architecture supports cooperative agreement creation when it can reach agreements with other parties trying to maximise only the social welfare.
17. *Consumer or provider orientation (S.1, S.3 and S.4)*: An abstract architecture is consumer-oriented if it carefully describes the behaviour of the consumer (or the party acting on his behalf) in the service trading process. Alternatively, it is provider-oriented if it carefully describes the behaviour of the provider (or the party acting on his behalf). Note that an abstract architecture may be both consumer and provider-oriented.
18. *Deployment options*: An abstract architecture may present several deployment options depending on their characteristics. Some examples of deployment are to integrate the architecture in the service provider or to implement an independent trader of services offering its trading services to several service providers or consumers.
19. *Assessment mechanisms (S.1, S.3 and S.4)*: The assessment mechanisms of an abstract architecture is the kind of information used in the architecture to evaluate the goodness of a proposal or agreement in relation to some criteria provided by the user [12]. For instance, the most usual assessment mechanism in service trading is utility functions.
20. *Forms of expressing information and preferences (S.1, S.2, S.3 and S.4)*: The preferences and the information managed about the service and the

parties can be expressed in different ways. Each abstract architecture may have their own way to express them, however, the most commonly used are to express them as constraints or as rules.

4 Analysis of Service Trading Architectures

Our goal is to apply the set of properties defined in the previous section to the most relevant abstract architectures. In this context, an abstract architecture is a specification that defines a set of elements (subsystems, components, interfaces, data types, or collaborations) for service trading and that can be applied for different domains and implemented with different technologies. Therefore, it is not the goal of this paper to analyse concrete architectures such as CREMONA [13].

Open Grid Services Architecture [5] is an abstract architecture for Grid systems and applications. The analysis of OGSA is based in [5] and other specifications developed by GGF (Global Grid Forum), which detail some aspects not fully described in that document, such as WS-Agreement [9]. The *discovery* employed is external and it is carried out by the so called *information services.* The architecture is *symmetric* for service consumer and provider as both of them may act as agreement initiators in WS-Agreement. There is no specific element to deal with *knowledge adaptation* during discovery, although the use of semantic-enabled discovery services could solve that problem. Concerning the *market observation*, it is achieved by using a subscription mechanism specified in the Grid Monitoring Architecture. Like discovery, both the *information query* and the *third party information* is developed by using the information services. The *agreement creation mechanism* employed in OGSA is the WS-Agreement protocol, although a negotiation protocol is also being developed and agreements for composed services can be created by using the *Execution Planning Services.* Concerning the *deployment*, OGSA is conceived to be deployed as independent services that are later used by higher-level applications. Finally, elements that support the decision-making such as the creation of a *world model* and the *types of information managed about the parties* together with the *assessment mechanisms* and the *forms of expressing information and preferences* are not in the scope of the architecture. This is also the case of the *proposals preselection*, although *Candidate Set Generator* could develop that function.

Semantic Web Service Architecture [14] describes an abstract reference architecture for semantic web service interoperability. In this architecture, the *discovery* issues are addressed from the perspective of semantic registries (*Matchmakers*).The *knowledge* management is a key point in this architecture expressed in the specification of different ontologies. In this context, despite the idea of market can be induced from this architecture, there is not an explicit element that actively reacts to different changes in the market (*Market observation*). This architecture is not *symmetric* due it is highly focused in the organisation that acts as service consumer and leaves the service provider as a comparatively simple systems that remain passive during the service trading process. The *information query* mechanism can be developed during the *engagement phase* in the *contract* *preliminaries* interaction. However, there is not specified an interaction with *third parties.*The interaction mechanisms related with *agreement creation* are based on abstract protocols described in FIPA Conversational Language. There is an explicit requirement for non-repudiation mechanisms during the *enactment* phase. Finally, though it is not specifically stated, this architecture is oriented toward non-cooperative scenarios.

Web Services Modelling Ontology Full [15] presents an abstract conceptual architecture for semantic web services and it is oriented to cross-organisational scenarios. It uses an *external discovery* based in the Web Service Architecture and it is symmetric for consumer and provider. It also supports *knowledge adaptation* by using semantic-based service descriptions. However, there is not explicit information about how to carry out a *market observation*, the *information query* nor *third party information*. Regarding the *information managed* about the parties, it uses service and trading information (e.g. supported choreographies) but it is not stated whether it can use information related to the parties. Like OGSA, the mechanisms to support decision-making are out of the scope of WSMO-Full. Therefore, neither the *world model*, the *capacity estimator* nor the *assessment mechanisms* are covered. The *agreement creation mechanisms* supported are specified through the so called *contract agreement choreographies*. WSMO-Full includes partial support for *decommitment* in the post-agreement choreography but the mechanism is not fully defined. It also partially supports *trading protocols* through contract agreement and post-agreement choreographies but it does not consider the specification of temporal constraints on them. However, WSMO-Full does not include any support for complex service trading elements such as a *notary* or the *creation of agreements for composed services*. The architecture seems conceived to operate in a non-cooperative agreement creation, although there is no explicit limitation in using it in a cooperative environment. Finally, as it is a conceptual architecture, it does not consider any *deployment* options.

Adaptive Service Grid [4] has been developed as an intent to create service providers that quickly adapt to business changes. In particular, the main goal is to achieve an efficient way of composing services to create more complex services with an added value. In the case of *symmetric* property, the elements that implement the provider-part and consumer-part of the system in ASG are not symmetric. The *discovery* is handled by the so called *DDBQuery* in a centred way through a semantic registry (with reasoning capabilities). In this way, though different ontologies handling are considered as part of the registry there is not an explicit *market* that is observed. Concerning the *world model*, this architecture specifies an element called *ServiceProfiling* that stores information about historic interactions with providers creating a relative model of the provider that is taken into account for the optimisation of the negotiation and selection of services. The *information managed* about parties is highly oriented to service; neither provider nor trading information are described in any of the processes. This approach leaves open the specific negotiation protocol used to establish the SLA for each service composed. However, WS-Agreement standard is specified as an implementation option. ASG can be applied to either cooperative

Table 1. Comparison of abstract architectures

	OGSA	SWSA	WSMO-Full	ASG
(1)	Yes (distributed)	Yes	Yes	No
(2)	No	Yes (ontologies)	Yes (ontologies)	Yes (ontologies)
(3)	Yes	No	No	No
(4)	Yes	No	Yes	No
(5)	Yes	Yes	No	No
(6)	Out of scope	Out of scope	Out of scope	Yes
(7)	Yes	No	No	No
(8)	Out of scope	Service, party	Service, trading	Service
(9)	Partial	Out of scope	No	No
(10)	WS-Ag	Yes, FIPA-CL based	Yes, through chor.	Yes (e.g. WS-Ag)
(11)	No	Yes	No	No
(12)	No	No	Partial	No
(13)	Yes	No	Out of scope	No
(14)	No	No	Lacks temporal constr.	No
(15)	Yes	No	No	Yes
(16)	Seems coop.	Seems non-coop	Seems non-coop	Both
(17)	Both	Consumer	Both	Chiefly Consumer
(18)	Independent services	Out of scope	Out of scope	Technologies
(19)	Out of scope	Out of scope	Out of scope	Out of scope
(20)	Out of scope	Semantic info	Semantic info	Semantic Info

or non-cooperative scenarios. Despite ASG describes the architecture of a composed services provider, from an architectural point of view this case is chiefly *service consumer oriented* because it just looks for atomic service providers to be composed. In ASG, the *deployment* possibilities are specified in terms of different development technologies and by identifying subsets of elements that are mandatory and other that can be optional.

5 Conclusions

From the analysis developed in Section 4, we can extract several conclusions: (i) The discovery process is well supported and most abstract architectures provides knowledge adaptation; (ii) Most abstract architectures do not cover elements to support the decision-making such as the world model; (iii) There is little support for the most advanced features of service trading such as the notary, the decommitment from established agreements and the trading protocols. Due to these lacks, some complex service trading scenarios cannot be completely achieved. Therefore, it may be interesting to develop new abstracts architecture to deal with those scenarios taking the set of properties obtained in this article as a starting point.

In summary, the contributions of this paper are: first, we obtain a set of properties of abstract service trading architectures based on an analysis of several service trading scenarios, and second, we use these properties to analyse and compare the most relevant abstract architectures for service trading. In so doing, we set the basis for the development of a method to select the service trading architecture most appropriate to the scenario where it is applied.

The future work is twofold. On the one hand, analysing additional service trading scenarios to identify the properties that an abstract architecture for service trading must have to successfully operate in them in order to define a method to select the architecture for each of them. On the other hand, we intend to extend the work to lower-level properties of non-abstract architectures so that they cover concrete technologies, protocols and algorithms.

References

1. Molina-Jiménez, C., Pruyne, J., van Moorsel, A.P.A.: The role of agreements in it management software. In: Architecting Dependable Systems III. (2004) 36–58
2. Ludwig, H., Gimpel, H., Dan, A., Kearney, R.: Template-Based Automated Service Provisioning - Supporting the Agreement-Driven Service Life-Cycle. In: ICSOC. (2005) 283–295
3. Ludwig, H.: A Conceptual Framework For Building E-Contracting Infraestructure. In Corchuelo, R., Wrembel, R., Ruiz-Cortes, A., eds.: Technologies Supporting Business Solutions. Nova Publishing (2003)
4. Laures, G., Jank, K.: Adaptive Service Grid Reference Architecture. `http://www.asg-platform.org` (2005)
5. Global Grid Forum, .: Open Grid Service Architecture. `http://www.ggf.org/documents/GFD.30.pdf` (2005)
6. Lee, J., Goodwin, R.: Ontology Management for Large-Scale E-Commerce Applications. In: DEEC. (2005) 7–15
7. Giunchiglia, F., Yatskevich, M., Giunchiglia, E.: Efficient Semantic Matching. In: ESWC. (2005) 272–289
8. Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Wooldridge, M., Sierra, C.: Automated Negotiation: Prospects, Methods and Challenges. Group Decision and Negotiation **10** (2001) 199–215
9. Andrieux, A., Czajkowski, K., Dan, A., Keahey, K., Ludwig, H., Nakata, T., Pruyne, J., Rofrano, J., Tuecke, S., Xu, M.: WS-Agreement Specification (2004)
10. Sandholm, T., Lesser, V.: Leveled commitment contracts and strategic breach. Games and Economic Behavior **35**(1) (2001) 212–270
11. Chung, J.Y., Bichler, M.: Service-oriented enterprise applications and Web service composition. Inf. Syst. E-Business Management **3**(2) (2005) 101–102
12. Wang, Y., Tan, K.L., Ren, J.: Towards autonomous and automatic evaluation and negotiation in agent-mediated internet marketplaces. Electronic Commerce Research **5**(3 - 4) (2005) 343–365
13. Ludwig, H., Dan, A., Kearney, R.: Cremona: An Architecture and Library For Creation and Monitoring of WS-Agreements. In: Proc. of the 2nd International Conference On Service Oriented Computing, ACM Press (2004)
14. Burstein, M., Bussler, C., Zaremba, M., Finin, T., Huhns, M.N., Paolucci, M., Sheth, A.P., Williams, S.: A Semantic Web Services Architecture. IEEE Internet Computing **9**(5) (2005) 72–81
15. Preist, C.: Agent Mediated Electronic Commerce Research At Hewlett Packard Labs, Bristol. SIGecom Exch. **2**(3) (2001) 18–28

An Ontological Approach for Translating Messages in E-Negotiation Systems

Víctor J. Sosa[1], Maricela Bravo[2], Joaquín Pérez[2], and Arturo Díaz[1]

[1] Centro de Investigación y de Estudios Avanzados (CINVESTAV)
cd. Victoria, Tamps, 87260, México
vjsosa@ieee.org, adiaz@cinvestav.mx
[2] Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET),
Cuernavaca, Mor., 62490, México
{vjsosa, mari_clau, jperez}@cenidet.edu.mx

Abstract. Traditional negotiation systems have been implemented using agent architectures, where agents communicate through the exchange of messages, based on particular language definitions implicitly encoded, using different implementations and meaning in their messages. Our approach focuses on solving the language heterogeneity problem between agents during a negotiation process, by incorporating an ontology-based translator solution, which is executed only when a misunderstanding occurs. We designed the translator architecture considering that agents involved in a negotiation process may be using similar languages, and not all exchanged messages will cause failures due to misunderstandings. We executed experiments in a Web-based electronic negotiation system, incorporating multiple agents with different language syntax and meaning. The experimental tests show that the proposed solution improves the continuity of the execution of negotiation processes, resulting in more agreements.

1 Introduction

Communications in electronic negotiation systems are crucial to achieve an agreement. A traditional electronic negotiation system is populated by software entities called agents, where agents communicate through the exchange of messages derived from a particular agent communication language (ACL), for example KQML [1] or FIPA ACL [2]. Although negotiation messages are based in one of these specifications, detailed syntax and meaning of such messages differ from one system to another depending on the developer's convenience, causing language heterogeneity. Recently there has been a growing interest in conducting negotiations over Internet, and constructing large-scale agent communities based on emergent Web standards. The challenge of deploying and integrating heterogeneous agents in open and dynamic environments is to achieve interoperability at the communication level, reducing misinterpretation of exchanged messages during negotiation processes.

An ACL allows an agent to share information and knowledge with other agents, or request the execution of a task. KQML was the first standardized ACL from the

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 213 – 221, 2006.

ARPA knowledge project. KQML consists of a set of communication primitives aiming to support interaction between agents. KQML includes many performatives of speech acts. Another standard ACL comes from the Foundation for Intelligent Physical Agents (FIPA) initiative. FIPA ACL is also based on speech act theory, and the messages generated are considered as communicative acts.

The objective of using a standard ACL is to achieve effective communication without misunderstandings, but this is not always true. Because, standards specify the semantics of communicative acts, but the software implementation is not explicitly defined, leaving developers to follow their own criteria. Furthermore, standard ACL specifications consider the incorporation of privately developed communicative acts. Thus, we consider that there is a problem of language heterogeneity when:

1. Agents involved in a negotiation process use different ACL for communication. To start a negotiation process, agents should use the same ACL to have a basic understanding.
2. Negotiation agents use the same ACL, but different versions or different implementations of such ACL.
3. Negotiation agents use the same ACL and same version, but messages generated by each agent have different syntax and/or meaning not based on explicit semantics, but on particular definitions implicitly encoded.

We have concentrated our work in the third case of language heterogeneity problem, and we have selected a translation approach based on the incorporation of a shared ontology. We implemented the ontology to explicitly describe negotiation messages in a machine interpretable form. The ontology represents the shared vocabulary that the translator uses during execution of negotiation processes for solving misunderstandings.

The rest of the document is organized as follows. In section 2, we present the related work. In section 3, we describe the translator architecture. In section 4, we present the design and implementation of the ontology. In section 5, we present the general architecture of the system for executing negotiation processes. In section 6, we describe the results of experiments. Finally in section 7, we present conclusions.

2 Related Work

According to Jürgen Müller [3], research in negotiation is organized in three classes: language, decision and process. We have concentrated our work in the language aspect of negotiation; in particular we are interested in analyzing research concerning the communication language interoperability between agents. In the revised works we have identified two trends in communications between negotiation agents, one is the generalized idea of using a standard, and the other is to provide mechanisms for solving heterogeneity. In particular, in this work we deal with the second trend. In this section we present the related work within this context.

Malucelli and Oliveira [4] stated that a critical factor for the efficiency of negotiation processes and the success of potential settlements is an agreement between negotiation parties about how the issues of a negotiation are represented and what this representation means to each of the negotiation parties. In [5] authors explain that interoperability is about effective use of systems´ services. They argue that the most important precondition to achieve interoperability is to ensure that the message sender and receiver share the same understanding of the data in the message and the same expectation of the effect of the message. S. Rueda [6] argues that the success of an agent application depends on the communication language, allowing agents to interact and share knowledge. Pokraev [7] discussed the problems of automating the process of negotiation. In this work he argues that there is a problem of lack of common understanding between participants in a negotiation, because messages are created by different actors and different meaning is given to the concepts used in them. In [8] authors explain that there are two important aspects of a negotiation process: communication between negotiation parties and decision-making. They state that communication deals with how to represent negotiator's requirements and constraints on a product and service and how to convey intentions by passing messages between negotiation parties. The lack of common language implementations represents a problem during the exchange of messages between heterogeneous systems, and this lack of standardization is known as interoperability problem [9].

In the above related works we can see that there is a common concern in communications in agent communities. Authors present the problem of lack of common understanding or the need for clarifying the meaning of concepts. But there is no common solution to the problem, not even a clear approximation. In this paper we present the incorporation of an ontology-based translator solution, which is executed only when a misunderstanding occurs.

3 Translator Architecture

We designed the translator architecture analyzing two possibilities. In figure 1, two architectural designs are shown. The architecture identified by letter *a*, was presented by Uschold [10]. This architecture was proposed to integrate different software tools,

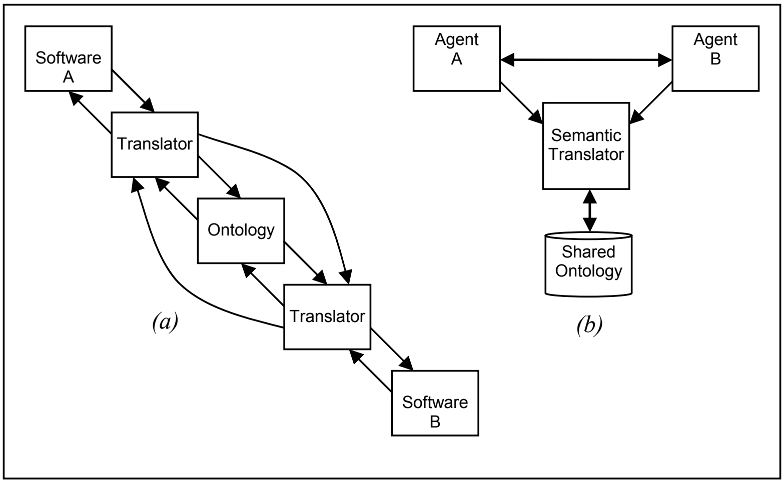


Fig. 1. Translator architectures

using an ontology as an *interlingua* to support translation between different languages. We consider that this is a good solution when systems use totally different languages, because communications are executed through the translator. The second architecture identified by letter *b* is our proposal. We designed this architecture considering that agents involved in a negotiation process may be using similar ACL, and not all messages generated will cause misunderstanding. Communications in our architecture are executed through an independent message transport, and only when agents need translation, the translator is invoked, reducing the number of translations.

3.1 Translator Functionality

The translator acts as an interpreter of different negotiation agents. In figure 2, we present the functionality of this module. The translator module first reads the input parameters, and then opens a connection to the Ontology to make queries. When the

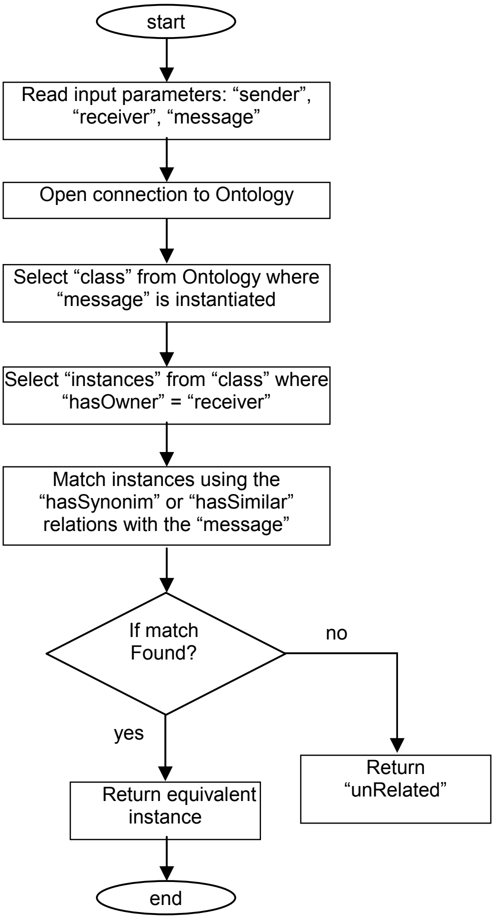

start
Read input parameters: “sender”, “receiver”, “message”
Open connection to Ontology
Select “class” from Ontology where “message” is instantiated
Select “instances” from “class” where “hasOwner” = “receiver”
Match instances using the “hasSynonim” or “hasSimilar” relations with the “message”
If match Found?
no
yes
Return equivalent instance
Return “unRelated”
end

Fig. 2. Translator functionality

connection has been established it narrows the search by selecting only the "class" of the incoming "message". It then searches for and retrieves all primitives that belong to the "receiver" agent in the same "class". And finally it makes a comparison to find a similar primitive, when this process ends it returns the equivalent primitive; in other case it returns a failure message.

A misunderstanding event occurs when an agent receiving a message, compares it to its own message knowledge base, and acknowledges that the received message is not in his base, and then invokes the translator to find the relation with its own messages.

For example, suppose that agents *A* and *B* initiate a negotiation process, using their own local ACL, sending messages over the message transport. If happens that agent *A* misunderstands a message from agent *B*, it invokes the translator module sending the message parameters (sender, receiver, message). The translator interprets the message based on the definitions of the sender agent and converts the message into an interlingua. Then the translator converts the interlingua representation to the target ACL based on the receiver agent definitions. Finally, the translator sends back the message to the invoking agent *A* and continues with execution of negotiation.

The translator is invoked only in the occurrence of a misunderstanding, assuring interoperability at run time.

4 Shared Ontology

Ontologies have been studied in various research communities, such as knowledge engineering, natural language processing, information systems integration and knowledge management. Ontologies are a good solution for facilitating shared understanding between negotiation agents. The principal objective in designing the ontology was to serve as an *interlingua* between agents during exchange of negotiation messages. According to Müller [3], negotiation messages are divided into three groups:

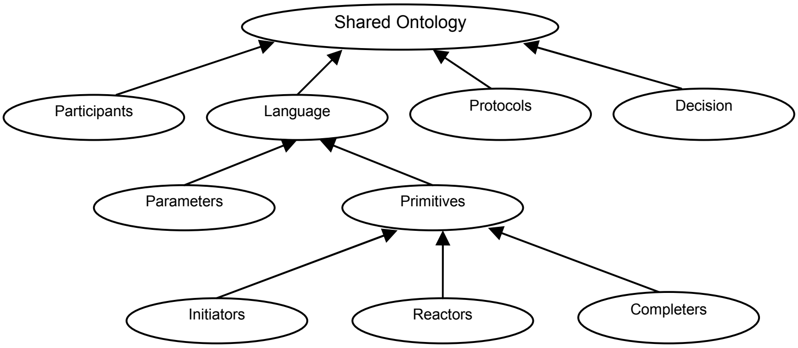


Fig. 3. General structure of the negotiation ontology

initiators, if they initiate a negotiation, reactors, if they react on a given statement and completers, whether they complete a negotiation. We selected this classification to allow the incorporation of new negotiation primitives from the local agent ACL. Figure 3 shows the general structure of our ontology.

We built the ontology using OWL as the ontological language, because it is the most recent development in standard ontology languages from the World Wide Web Consortium (W3C)[1]. An OWL ontology consists of classes, properties and individuals. We developed the ontology using Protégé [15, 16], an open platform for ontology modeling and knowledge acquisition. Protégé has an OWL Plugin, which can be used to edit OWL ontologies, to access description logic reasoners, and to acquire instances of semantic markup.

5 Implementation of the Negotiation System

The general architecture for the execution of negotiation processes is illustrated in figure 4. This architecture has the following elements: the matchmaker module, the negotiation process module and the translator module. The matchmaker module is continuously browsing buyer registries and seller descriptions, searching for coincidences. The negotiation process module controls the execution of negotiation processes between multiple agents according to the predefined protocols.

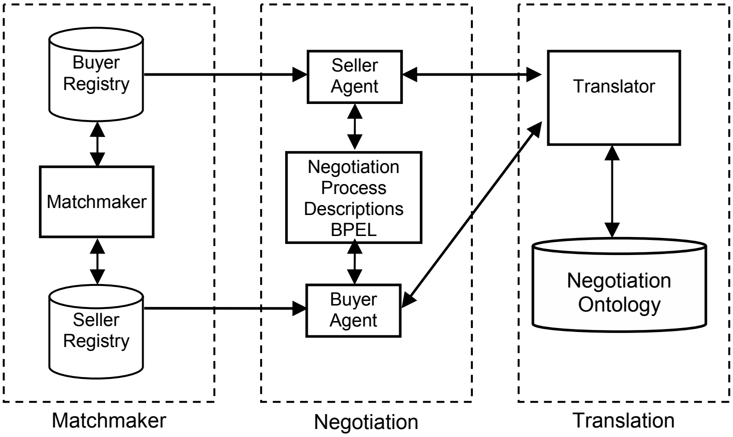


Fig. 4. Architecture of the prototype

The seller and buyer agents are the representative software entities used by their respective owners to program their preferences and negotiation strategies. For example, a seller agent will be programmed to maximize his profit, establishing the lowest acceptable price and the desired price for selling. In contrast, a buyer agent is seeking to minimize his payment.

[1] http://www.w3.org

The translator module is invoked whenever the agent misunderstands a negotiation message from another agent. The translator module was implemented using Jena[2], a framework for building Semantic Web applications. It provides a programmatic environment for OWL, including a rule-based inference engine. For the description and execution of negotiation processes, we used BPEL4WS. BPEL4WS defines a model and a grammar for describing the behavior of a business process based on interactions between the process and its partners. BPEL4WS represents a convergence of the ideas in the XLANG and WSFL specifications. Both XLANG and WSFL are superseded by the BPEL4WS specification. The interaction with each partner occurs through Web service interfaces, and the structure of the relationship at the interface level is encapsulated in what we call a partner link. The BPEL4WS process defines how multiple service interactions with these partners are coordinated to achieve a business goal, as well as the state and the logic necessary for this coordination.

6 Experimental Results

For the execution of experiments we first designed various negotiation agents with different language definitions. On designing the negotiation agents, we identified three core elements, strategies, the set of messages and the protocol for executing the negotiation process. The requirements for these elements were specified as follows:

1. Strategies should be private to each agent, because they are competing and they should not show their intentions.
2. Messages should be generated privately.
3. The negotiation protocol should be public or shared by all agents participating, in order to have the same set of rules for interaction. The negotiation protocol establishes the rules that agents have to follow for interaction.

We then populated the ontology using the agent language definitions. And finally we executed negotiations using the system described in section 5. The negotiation experiments were executed in two phases. The first execution tested the interaction between agents, incorporating messages with different syntax, without the translator. For the second execution we used the same scenario, but enabled the translator module. The results of these experiments were registered in a log file. Table 1 shows the results of both cases.

The first execution results showed that there were some negotiations that ended the process with no agreement. This was due to the private strategies defined inside the agents. But there were some negotiation processes that ended due to lack of understanding of negotiation messages.

The second phase results showed a reduction in the number of negotiations finished by lack of understanding, which does not mean that the incorporation of a translator module will ensure an agreement; but at least, the negotiation process will continue executing.

[2] http://jena.sourceforge.net

Table 1. Negotiation results

Last price	Max pay	Rounds	Qty	Final price	1st execution	2nd execution
$ 1,750.00	$ 849.00	12	847	$ -	no offer	no offer
$ 774.00	$ 1,760.00	3	887	$ 1,674.00	offer accepted	offer accepted
$ 1,788.00	$ 128.00	12	1660	$ -	no offer	no offer
$ 1,058.00	$ 110.00	12	1270	$ -	no offer	no offer
$ 694.00	$ 938.00	10	950	$ 894.00	offer accepted	offer accepted
$ 761.00	$ 77.00	12	1475	$ -	no offer	no offer
$ 1,940.00	$ 2,233.00	10	570	$ 2,140.00	offer accepted	offer accepted
$ 621.00	$ 446.00	12	56	$ -	no offer	no offer
$ 1,008.00	$ 1,235.00	10	30	$ 1,208.00	offer accepted	offer accepted
$ 114.00	$ 704.00	7	8	$ 614.00	offer accepted	offer accepted
$ 1,837.00	$ 2,199.00	9	53	$ 2,137.00	offer accepted	offer accepted
$ 1,665.00	$ 2,047.00	9	56	$ 1,965.00	offer accepted	offer accepted
$ 1,377.00	$ 1,783.00	8	31	$ 1,777.00	offer accepted	offer accepted
$ 1,920.00	$ 286.00	12	81	$ -	no offer	no offer
$ 172.00	$ 1,553.00	2	41	$ 1,172.00	offer accepted	offer accepted
$ 980.00	$ 1,541.00	2	67	$ -	**not understood**	offer accepted
$ 1,826.00	$ 2,464.00	2	99	$ -	**not understood**	offer accepted
$ 1,276.00	$ 500.00	2	43	$ -	**not understood**	no offer
$ 1,500.00	$ 1,108.00	2	110	$ -	**not understood**	no offer
$ 1,400.00	$ 1,520.00	3	4	$ -	**not understood**	offer accepted

Figure 5 shows a comparison for the two phases executed. The second phase shows the elimination of the “not understood” occurrence.

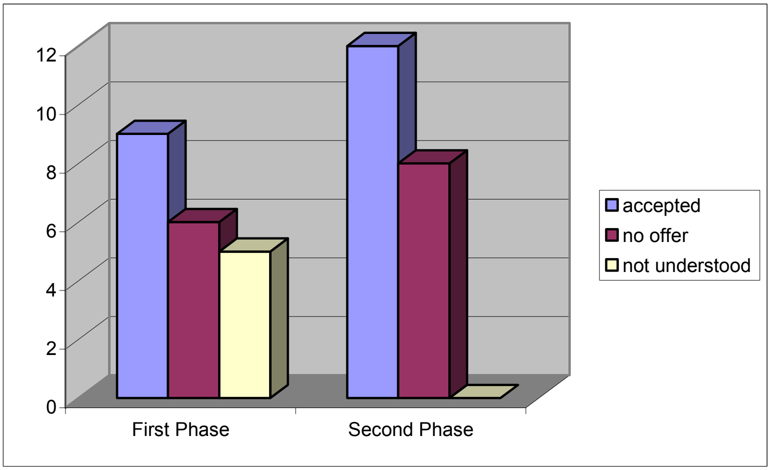


Fig. 5. Graphical comparison for the execution of experiments

7 Conclusions

In this paper we have presented a translator architecture based on the incorporation of a shared ontology, to address the problem of language heterogeneity. We implemented the ontology to explicitly describe negotiation messages in a machine interpretable form. The ontology represents the shared vocabulary that the translator uses during execution of negotiation processes for solving misunderstandings at run time. In particular, we implemented an efficient translator architecture, which translates only when is necessary. We evaluated the ontology in the target application, and described the system architecture into which the negotiation processes are executed. We believe that language interoperability between negotiation agents is an important issue that can be solved by incorporating a shared ontology. The experimental tests showed that the proposed architecture improves the continuity of the execution of negotiation processes, resulting in more agreements.

References

1. T. Finning, R. Fritzon, and R. McEntire: KQML as an agent communication language, in *Proceedings of the 3rd International Conference on Information and Knowledge Management*, November 1994.
2. FIPA – Foundation for Intelligent Physical Agents. FIPA Specifications, 2003; available at http://www.fipa.org/specifications/index.html.
3. H. J. Müller, , Negotiation Principles, *Foundations of Distributed Artificial Intelligence*, in G.M.P. O´Hare, and N.R. Jennings, New York: John Wiley & Sons, 1996.
4. A. Malucelli, and E. Oliveira, Towards to Similarity Identification to help in the Agents' Negotiation, *Proceedings of 17th Brazilian Symposium on Artificial Intelligence*, São Luis, Maranhão, Brazil, 2004.
5. S. Pokraev, M. Reichert, M. Steen and R. Wieringa, Semantic and Pragmatic Interoperability: A Model for Understanding, *Proceedings of the Open Interoperability Workshop on Enterprise Modelling and Ontologies for Interoperability*, Porto, Portugal, 2005.
6. Sonia V. Rueda, Alejandro J. García, Guillermo R. Simari, Argument-based Negotiation among BDI Agents, *Computer Science & Technology*, 2(7), 2002.
7. S. Pokraev, Z. Zlatev, R. Brussee, P. van Eck, Semantic Support for Automated Negotiation with Alliances, *Proceedings of the 6th International Conference on Enterprise Information Systems*, Porto, Portugal, 2004.
8. H. Li, C. Huang and S. Y.W Su, Design and Implementation of Business Objects for Automated Business Negotiations, *Group Decision and Negotiation*, Vol. 11; Part 1, pp. 23-44, 2002.
9. S. Willmott, I. Constantinescu, M. Calisti, Multilingual Agents: Ontologies, Languages and Abstractions, *In Proceedings of the Workshop on Ontologies in Agent Systems, Fifth International Conference on Autonomous Agents*, Montreal, Canada, 2001.
10. Uschold, M. and King M., Towards a Methodology for Building Ontologies, *Workshop on Basic Ontological Issues in Knowledge Sharing*, 1995.
11. J. Gennari, M. Musen, R. Fergerson, W. Grosso, M. Crubézy, H. Eriksson, N. Noy, and S. Tu: The evolution of Protégé-2000: An environment for knowledge-based systems development, *International Journal of Human-Computer Studies*, 58(1): 89-123, 2003.
12. H. Knublauch: An AI tool for the real world: Knowledge modeling with Protégé, *JavaWorld*, June 20, 2003

Detecting Frauds in Online Advertising Systems

Sanjay Mittal[1], Rahul Gupta[1], Mukesh Mohania[2], Shyam K. Gupta[1],
Mizuho Iwaihara[3], and Tharam Dillon[4]

[1] Dept of Computer Science, I.I.T. Delhi, Hauz Khas, New Delhi, India
[2] IBM India Research Lab, I.I.T. Delhi, Hauz Khas, New Delhi, India
[3] Dept of Social Informatics, Kyoto University, Kyoto, Japan
[4] Faculty of Information Technology, University of Technology, Sydney, Australia

Abstract. Online advertising is aimed to promote and sell products and services of various companies in the global market through internet. In 2005, it was estimated that companies spent $10B in web advertisements, and it is expected to grow by 25-30% in the next few years. The advertisements can be displayed in the search results as sponsored links, on the web sites, etc. Further, these advertisements are personalized based on demographic targeting or on information gained directly from the user. In a standard setting, an advertiser provides the publisher with its advertisements and they agree on some commission for each customer action. This agreement is done in the presence of Internet Advertising commissioners, who represent the middle person between Internet Publishers and Internet Advertisers. The publisher, motivated by the commission paid by the advertisers, displays the advertisers' links in its search results. Since each player in this scenario can earn huge revenue through this procedure, there is incentive to falsely manipulate the procedure by extracting forbidden information of the customer action. By passing this forbidden information to the other party, one can generate extra revenue. This paper discusses an algorithm for detecting such frauds in web advertising networks.

1 Introduction

In the increasingly wired world of today, the importance of web advertising can hardly be over-emphasized for marketing goods and services on a global platform. It is undoubtedly one of the best media to target customers and influence brand results. Currently having a market size of over $10 billion, Web advertising is projected to be a $17.3 billion business in 2007 [14]. Also, worldwide B2B Internet commerce is nearly $8.5 trillion, and B2C e-commerce market is estimated to be $133 billion [15]. A large part of this revenue comes from the "Search Advertising". Search Advertising has been embraced by advertisers due to its innate relevancy, the simplicity of the results and because advertisers can determine more precise response rates. In such a large commercial market, there is always a chance of some fraud to exploit the market. Web advertising is also believed to be susceptible always to some form of fraud. International e-commerce continues to be a far higher risk, with order rejection and fraud rates about three times higher than the overall rate [16]. The number of registered cases of fraud in Web Advertising has witnessed a tremendous increase in the last few years. $2.8 billion in e-commerce revenues was lost to fraud in 2005. The

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 222 – 231, 2006.

consequences are deleterious for the industry. It was seen that 3.9% of all e-commerce orders declined on the suspicion of fraud in 2005 [16]. Various kinds of frauds, such as hit inflation, hit shaving and click fraud, exists in web advertisement. The detection of the same becomes a vital problem in the growing internet advertising industry.

In this paper, we address one particular type of fraud that exists in search advertising. In this scenario, advertisers provide their advertisements and a related set of keywords to a publisher (a search engine), and they agree on a commission for some customer action(s). This agreement takes place in the presence of a middle party, called, the Internet Advertising commissioner. In return of the commission paid by advertisers, the publisher displays the advertisers' links in its search result. Since each player in this scenario can earn huge revenue through this procedure, there is incentive to falsely manipulate the procedure by extracting forbidden information of the customer action. By passing this forbidden information to any other party, one can possibly generate extra revenue. We probe a possible way to find such kind of fraud in this framework. We propose an algorithm to detect frauds in a web (i.e. online) advertising network, and illustrate the method with an example. We also show the performance results of our proposed algorithm.

The rest of the paper is organized as follows. The related work is outlined in Section 2. The problem is described formally in Section 3. The solution for fraud detection in the web advertising scenario is described in Sections 4. The experimental results are discussed in Section 5. Finally, we conclude the paper in Section 6.

2 Related Work

Guidance in formulating WWW advertising policy with respect to consumer attitudes has been provided in [7]. Realizing the lack of standardization in web advertising model, [8] presents different pricing models that may be used in the current scenario. Many click-through payment programs have been established on the web, by which (the webmaster of) a target site pays a referrer site for each click through that refers to the target [1, 2, 3 ,4]. In [2] a hit inflation attack on pay-per click web advertising is introduced which is virtually impossible for the program provider to detect conclusively. The authors argue that their attack is impossible to detect. However in [3], a solution to the problem described in [2] is proposed. It uses streaming rule algorithm to detect fraud presented. Paper [11] introduces the problem of finding duplicates in data streams to detect fraud in web advertising. More importantly, the 2 most dominant forms of advertiser supported websites that exist today are sponsored content sites and entry portal sites, that provide search and directory features to web browsers [8]. The practice of sponsored search advertising - where advertisers pay a fee to appear alongside particular Web search results - is now one of the largest and fastest growing sources of revenue for Web search engines [9]. In [10], the authors address certain issues present in search engine advertising like placement of links, safeguards search engines can and can't offer, identifying click fraud patterns, etc. Even though fraud in search advertising has profound implications on web advertising businesses, it has not been adequately addressed so far. We propose a way to detect malicious collaborations that are possible in such a scenario, and possess potential threat to the integrity of e-commerce industry.

3 The Problem Statement

This paper addresses the problem of how to detect a possible malicious collaboration between an advertiser and a search engine (publisher) in a particular Web Advertising scenario, wherein a list of advertisers officially collaborate with a search engine to display their links on the search result in exchange of appropriate revenue. But, under such a malicious collaboration, the search engine illegally but secretively discloses forbidden information about user's actions to one particular advertiser, for him to gain competitive business advantage conveniently.

3.1 Web Advertising Networks Framework

The Figure 1 describes the Web Advertising Networks framework. In this section, we describe the functionalities of each component.

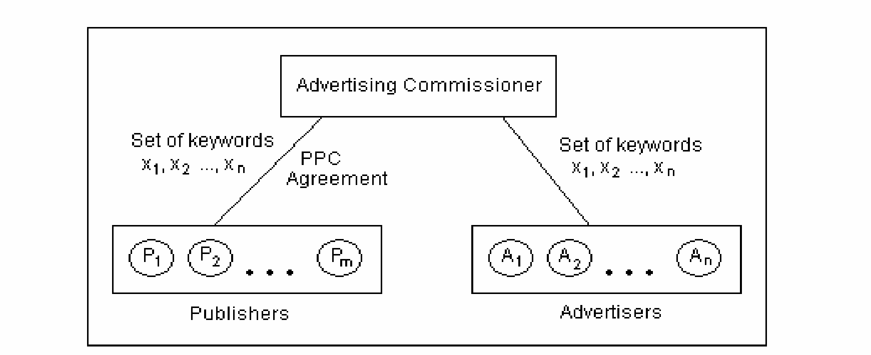


Fig. 1. Web Advertising Networks Framework

Advertisement Publishers (P_1, P_2, … P_m): These are independent parties that promote the products or services of an advertiser, based on a Pay-per-click agreement, in exchange for a commission on leads or sales. A publisher displays advertisements, text links, or product links on its Web site, or in email campaigns, or in search listings and is paid a commission by the advertiser when a visitor takes a specific action.

Advertisers (A_1, A_2, … A_n): Advertisers place advertisements and links related to their products and services on other Web sites (publishers) and, through an Advertising Commissioner, pay those publishers a commission for leads or sales that result from their sites. Each advertiser A_i provides a list of keywords $\{x_1, x_2, \ldots x_n\}$ to the commissioner, that best describes his/her products and services.

Advertising Commissioner: Being the middle party in this scenario, Advertising Commissioner is responsible for displaying advertisements of advertisers at other websites (publishers) such that the advertisers can achieve maximum gain from the advertisements. There is no such direct link between Advertisers and Publishers. Advertising commissioner is responsible for all legal issues that may arise in future. The Commissioner, on the behalf of the advertiser, provides the keywords $\{x_1, x_2, \ldots x_n\}$ and a corresponding list of advertisers to each publisher P_j. According to the agreement, the publisher P_j displays the links of advertisers whenever any user searches using keywords x_j.

3.2 Illustrative Example

To study the problem closely and to motivate the reader, we narrow down our analysis to a specific example of this type of fraud, which reveals the number of issues in detecting such frauds. We later generalize the problem and propose an algorithm to detect this kind of fraud. Consider a search engine XYZ.com (the publisher) and suppose some airline companies A_1, … A_n have a Pay-Per-Click (PPC) advertisement contract with XYZ.com. This type of agreement between XYZ.com and Airline companies is carried out in the presence of Advertising commissioner. According to the agreement, whenever there is any enquiry regarding Airline reservation at XYZ.com, links of these airlines' web sites are displayed. The exact placement of the ad links is based on how much one bids against his competitors; the more one bids, the higher in position his link appears. Without loss of generality, let's assume that links of A_1, … A_i are displayed at positions 1, …., i respectively, from the top, which is in accordance with the amount they pay.

Now, consider a hypothetical case, where a user wants to enquire about the airfare from New Delhi to Mumbai. For this purpose, the user visits the site XYZ.com and types the phrase – 'Airfare from New Delhi to Mumbai'. Now as per the advertisement contract links are displayed in order. It is fair to assume that user will click on the link as per the order in which they are displayed. So initially, we suppose that the user goes to A_1.com, enquires about the cheapest fare and then switches over to A_2.com, and then moves further. After getting the required airfare from all these airlines he will analyze the fares and will opt for the one who offers the cheapest fare.

Till here everything seems to be fine and thus, the one which offers competitive rates should have maximum market share. But, a possibility which lies here is that A_i (say), wanting to maximize its revenue from online booking, may try to collaborate with XYZ.com in an unofficial way. According to this probable unofficial contract, XYZ.com will display the link of A_i at the i^{th} position only, but it will provide the links of all other airline sites which the user have visited already, that is prior to visiting Ai's site, to A_i.

Why will XYZ.com do so? Generally the honest and ethical sites won't do this, as this may ruin their market image in future. But in such a competitive world everybody wants to maximize his/her profit at whatever cost. In lieu of links, (visited by user before coming to A_i.com) XYZ.com can charge a good amount of money from A_n.

What will A_i do after getting links from XYZ.com? On getting the links visited by the user, A_i will surf those websites and get the fares given by them. After knowing the fares offered by them, A_i.com will offer the lowest fare among these. So, this is likely to lead to an increase in the online revenue of A_i and decline in other airlines' revenue. Thus, A_i can generate much more business. The process of getting Airfare from A_1,…, $A_{(i-1)}$ if done manually by A_i would take a large amount of time; so we suppose that such a process is automated as shown below.

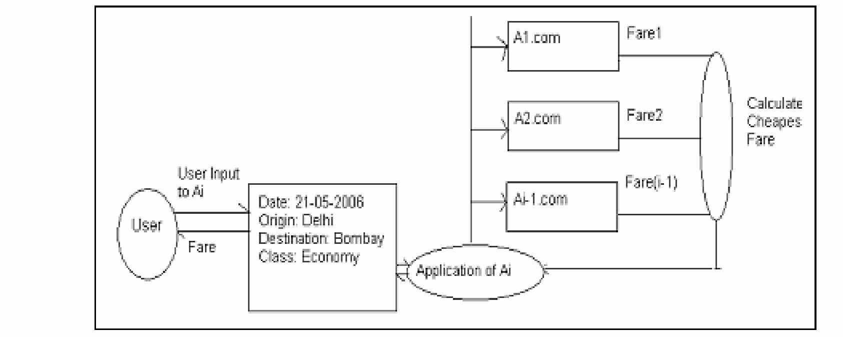


Fig. 2.

Thus, we see that collaboration of two websites in such a setting can lead to the decline in the business of the other. This type of fraud can exist in other similar scenarios where the user makes purchases with the help of a search engine. In this specific example, A_1 or other airlines may realize that they are not gaining revenue generated by referrals from XYZ.com despite offering competitive rates each time.

4 Fraud Detection in Web Advertising

In this section we describe the proposed solution of detecting a malicious collaboration, which leads to frauds, in web advertising networks. There are many interesting issues involved in this problem. We discuss these issues and illustrate how we have taken these issues into consideration while describing the fraud detection algorithm.

4.1 Analysis of Problem

Again, the question is how to detect malicious collaboration in the advertisement network as discussed in previous sections? The only entity that can help in detecting such an association is the Internet Service Provider (ISP), through which the customer logs on to the Internet. The solution of this problem requires analyzing the stream of HTTP requests that have been made by the customers in a specific time interval. ISP maintains the log of each customer's internet activity i.e., record of each page visited by customer, time spent there, from which IP he or she has come, etc. The part of ISP log that is accessible to the Advertising Commissioner is of the form $[(a_1, t_1), (a_2, t_2)\ldots(a_n, t_n)]$, where each tuple (a_i, t_i) denotes that page 'a_i' is requested at time 't_i'. So, ISP Log will contain all the web pages requested in the time period t_1 to t_n.

Revisiting the above example step-wise:

- User visits XYZ.com (say, S_1); queries about airfare from Delhi to Mumbai
- XYZ.com displays the results as $A_1\ldots A_i,\ldots,A_n$
- User clicks on the links in the order in which they are displayed and gets airfare offered by A_1, A_2 ……
- User clicks on A_i and enquires about airfare from Delhi to Mumbai.

- XYZ.com gives the links already visited by the user to A_i i.e. (A_1…$A_{(i-1)}$)
- A_i visits each of these links (A_1…$A_{(i-1)}$) through an automated application, and gets to know the relevant fare offered by them
- A_i dynamically changes its fare and offers a fare which is lower than all the other fares.

Now, if we consider visiting links as HTTP requests, then our ISP log would look like, [(S_1, T_1)…(A_1, T_2)...(A_i, T_i)…(A_1, T_p)…($A_{(i-1)}$, T_q)]. Here, dots (…) represent HTTP requests from other internet users who are accessing internet from the same ISP. There may be many HTTP requests to the ISP at the same time.

4.2 Analysis of Http Stream Requests

Considering the running example, the owner of A_1 airlines can investigate the collaboration between S_1 and A_i, if he realizes that he is not earning any business through S_1. With the help of the Commissioner, the stream of HTTP requests for a particular period of time can be analyzed. There exists forward association between some elements.

The association ($S_1 \rightarrow A_1$) tells that page A_1 is requested after page S_1. So, if A_1 suspects a malicious collaboration between S_1 and A_i, then he would be interested in finding all the occurrences of $S_1 \rightarrow A_1$ and within some span of this occurrence, he would analyze the other stream requests. The second forward association is of type ($A_1 \rightarrow A_i$) which tells us that page A_i is visited after A_1. Now, if there is a visit on A_1 again after $A_{i,}$ and A_1 is losing revenue in this case, then there is a possibility that such an association exists.

In terms of semantics, ($S_1 \rightarrow A_1$, $A_1 \rightarrow A_i$) → ($A_i \rightarrow A_1$) is the association. Whenever such a condition exists, and A_1 has not got business, then there lies the possibility of a fraud. Because, in an otherwise normal scenario, if an $A_1 \rightarrow A_i$ and $A_i \rightarrow A_1$ association exists, then that would imply that the user has revisited A_1 since A_1 is probably offering the cheapest fare, and hence A_1 should be the one selling the ticket and gaining business.

We will analyze the HTTP stream requests and find out all the associations that exist in the stream. We will consider the associations between S_1 and A_1 in a time span of sp_1 (Page A_1 is requested after S_1 within time span of sp_1). For finding the association between A_1 and A_i ($A_1 \rightarrow A_i$) we will consider a different time span sp_2, because the time between a click on A_1 and A_i will be more than that of sp_1. Note that in this time period, the user will enquire about the fare from sites A_1, … A_{i-1}, and in the end he will click on A_i). Now to find the association of type $A_n \rightarrow A_1$ we will consider time span of sp_3 because page A_4 is requested automatically by the automated script run by A_4 (so it's probable that sp_3 will be less than sp_1).

So in general, on getting the ISP log, we take the first element and consider it as a base element. We set time stamp of this first element to zero. Time stamp of other elements are modified on the basis of the first element. All these new time stamps are converted into seconds. For generic case, the task is to find the association of type -

$[(X_1 \rightarrow X_2),(X_2 \rightarrow X_3)] \rightarrow (X_3 \rightarrow X_2)$ where,

X_1 = Publisher suspected by owner of X_2

X_2 = Advertiser who is losing revenue

X_3 = Competitor who may be in malicious collaboration with the publisher
sp_1 = Maximum Time Period between visiting X_1 and X_2 by user
sp_2 = Maximum Time Period between visiting X_2 and X_3 by user
sp_3 = Time in which automated application of X_3 can visit other sites
$count_1$ = counts the no. of occurrences of type $[(X_1 \rightarrow X_2),(X_2 \rightarrow X_3)]$
$count_2$ = counts the no. of occurrences of type $[(X_1 \rightarrow X_2),(X_2 \rightarrow X_3)] \rightarrow (X_3 \rightarrow X_2)$

4.3 Algorithm

Here, we present a formal representation of our algorithm which we wish to propose. The inputs to the algorithm include the ISP log that is available to the Advertising Commissioner, sites X1, X2 and X3, and the time spans sp1, sp2 and sp3. The algorithm scans the ISP log four times, each time marking the elements which form the associations we need to detect in order to suspect the malicious collaboration. We maintain the count of the number of times particular associations are detected, so that we can use it for further analysis. The algorithm given below finally gives the total number of occurrences of the association $[(X_1 \rightarrow X_2),(X_2 \rightarrow X_3)] \rightarrow (X_3 \rightarrow X_2)$ that it detects through analysis of the ISP log.

```
begin
For each element x in the ISP log
   If x = $X_1$, then
      Mark x as [$X_1$]
For each element x in the ISP log
   If x = [$X_1$], then
      For each element y within time span $sp_1$ of x
            If y = $X_2$, then
            Mark y as [$X_2$]
For each element x in the ISP log
   If x = [$X_2$], then
          For each element z within time span $sp_2$ of x
            If z = $X_3$, then
                    Mark z as [$X_3$]
                    Increase the $count_1$ by1
For each element x in the ISP log
   If x = [$X_3$], then
          For each element w within time span $sp_3$ of x
               If w = $X_1$, then
                    Increase the $count_2$ by 1
end
```

Applying the above algorithm in running example,

- Find all the forward associations of form $S_1 \rightarrow A_1$, within a time span sp_1.
- Mark all such occurrences of A_1 with $[A_1]$.
- Find the forward associations of type $[A_1] \rightarrow A_i$ within a time span sp_2.
- Mark all such occurrences of A_i with $[A_i]$.
- Now we have all the occurrences of type $(S_1 \rightarrow A_1, A_1 \rightarrow A_i)$.

- Find forward associations of type $[A_i] \rightarrow A_1$ within time span sp_3 and count the number of such occurrences.
- If such associations exist, then there is a possibility of some malicious collaboration between S_1 and A_i.

There may be some cases in which XYZ.com, A_1.com and A_i.com are visited independently, and our algorithm may have found out such occurrences and counted them as well. But such number would be very small and there would be some error in our calculation. Moreover, this error does not matter as A_1 is not interested in counting total number of cases where this association exists. A_1 is only interested in whether or not there exists any collaboration between XYZ.com and A_i. So, by analyzing the result of this algorithm, he can infer where or not there exists such collaboration.

In the next section we will discuss some of the issues involved in the performance of this algorithm. We further show the experimental results obtained by implementing this algorithm and finally conclude in the last section.

4.4 Issues Involved

4.4.1 Time Span

We have considered 3 parameters in our algorithm, time span sp_1, sp_2 and sp_3. These parameters represent the time span within which the next click will be made by the user on the links of the search result, and we find the association rules [5] within these time spans. These parameters depend upon the time taken by the advertiser to answer the query of the user about a specific product or service. Thus, these parameters are domain dependent.

For example, the time span sp_2 depends upon the position of link of the malicious advertiser. The time span sp_3 depends on the time taken by the automated application of the malicious advertiser. To support our experimentation results and assumptions of the parameters we have assumed in the simulation, we have referred to survey.

4.4.2 Position of Link of Advertiser in Search Listing

In this scenario, where a search engine and advertisers can have a malicious collaboration, the best position for the malicious advertiser in the search result is a matter of choice. The best position for any advertiser in the search result not only depends on the products or services advertiser is offering but it also depends on the user who is going to use these products and services. Lot of surveys have been done on the search listing results and users preferences and demographics [12]. However, we have presented a generic algorithm which is independent of these factors. Also, domain dependent factors like these can be easily incorporated into it.

5 Experimental Results

We implemented our algorithm on simulated ISP data which we generated in the form of random http requests. Considering each possible position of link of the malicious advertiser in the search engine one at a time, we generated sets of ISP data by simulating http requests. Once we obtained this data for each position, we ran our algorithm on it and varied the input parameters. Our program was run on a sample of

200 users. There are 6 input parameters to our algorithm. The 3 parameters, sp_1, sp_2 and sp_3, which are time spans, affect the efficiency of our algorithm. Among these time spans, sp_2 is the most significant one. This represents the time between clicking the link of the advertiser who is losing revenue (A_1, in our example case) and the link of malicious advertiser. For instance, if the link of the malicious advertiser is placed at the bottom, then this time span sp_2 must be sufficiently larger than the other two parameters.

Implementing the above running example, we assumed that 7 advertisers have a PPC agreement with a publisher P. Keeping the time spans sp_1 and sp_3 same each time, we vary sp_2. The efficiency of algorithm comes out to be higher for higher value of sp_2.

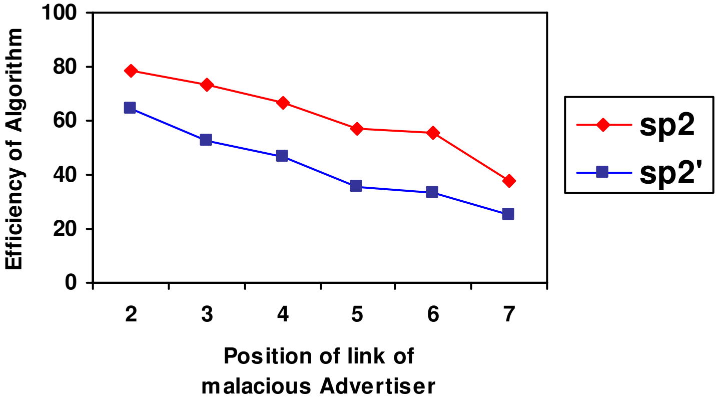


Fig. 3.

Performance of the algorithm is sketched in the above figure and it is seen that different values of sp_2 gives difference result. In the worst case scenario, when time span sp_2 is very small, our algorithm was able to identify 25% of the above associations which exposed the malicious collaboration between the publisher and advertiser. Efficiency of the algorithm can be increased if time span sp_2 is made appropriately large.

The experimental results confirm that if domain dependent factors like time spans are chosen appropriately and fed into the algorithm proposed above, such malicious collaborations as defined in this framework can possibly be detected.

6 Conclusions

Web Advertisement business is booming rapidly today, but, at the same time, it is also believed to be susceptible to some forms of fraud. This paper has addressed to detect one kind of a fraud that may exist in web advertising networks. We have proposed an algorithm to detect the possible malicious collaboration between a search engine (publisher) and a particular advertiser, wherein forbidden information regarding custommers' actions are provided secretively to that advertiser to conveniently gain advantage over its competitors. Such a fraud can exist in any kind of scenario where purchases are made online through a search engine, and a list of advertisement companies compete to gain more business. The analysis of our detection approach shows that it is indeed possible for a concerned party, say another advertisement company, to detect such a fraud with the help of the Advertising Commissioner in place.

References

1. Michael K. Reiter, Vinod Anupam, Alain Mayer. “Detecting Hit Shaving in Click-Through Payment Schemes”. In *Proceedings of the 3rd USENIX Workshop on Electronic Commerce*, pages 155-166, Boston, USA, 1998.
2. Vinod Anupam, Alain Mayer, Kobbi Nissim, Benny Pinkas, Michael K. Reiter. “On the security of pay-per-click and other Web advertising schemes”. In *Proceedings of the 8th WWW International World Wide Web Conference*, pages 1091-1100, Toronto, Canada,1999.
3. Ahmed Metwally, Divyakant Agrawal, Amr El Abbadi. “Using Association Rules for Fraud Detection in Web Advertising Networks*”*. In *Proceedings of the 31st International Conference on Very Large Databases (VLDB),* Trondheim, Norway, 2005.
4. Vinod Anupam, Alain Mayer. “Secure Web scripting”. *IEEE Internet Computing 2 (6),* pages 46-55, 1998
5. Rakesh Agrawal, Ramakrishnan Srikant. “Fast Algorithms for Mining Association Rules in Large Databases”. In *Proceedings of the 20th International Conference on Very Large Data Bases,* pages 487-499, Santiago, Chile, 1994.
6. Vinod Anupam, Alain Mayer. “Security of web browser scripting languages: Vulnerabilities, Attacks and Remedies”. In *Proceedings of the 7th USENIX Security Symposium,* San Antonio, Texas, USA, 1998.
7. Mary Ellen Gordon, Kathryn De Lima-Turner. “Consumer attitudes towards Internet advertising - A social contract perspective”. *International Marketing Review 14 (5)*, pages 362-375, 1997.
8. Donna L. Hoffman, Thomas P. Novak. “Advertising Pricing Models for the World Wide Web”. In *Internet Publishing and Beyond: The Economics of Digital Information and Intellectual Property*, Cambridge: MIT Press, 2000.
9. Juan Feng, Hemant K. Bhargava, David M. Pennock. “Implementing Sponsored Search in Web Search Engines: Computational Evaluation of Alternative Mechanisms”. *INFORMS Journal on Computing*, 2006.
10. Catherine Seda. “Search Engine Advertising: Buying Your Way to the Top to Increase Sales”. New Riders Press, 2004.
11. Ahmed Metwally, Divyakant Agrawal, Amr El Abbadi. “Duplicate Detection in Click Streams”. In *Proceedings of the 14th WWW International World Wide Web Conference*, pages 12-21, 2005.
12. iProspect Search Engine User Attitudes Survey, 2004. Available at URL http:// www.iprospect.com/premiumPDFs/iProspectSurveyComplete.pdf /
13. The Carmel Group Market Research (http://www.carmelgroup.com)
14. Gartner, Jupiter Research (http://www.jupiterresearch.com)
15. Cybersource Reports (http://www.cybersource.com)
16. ZDNet Research Study (http://news.zdnet.com)

An Improved Web System for Pixel Advertising*

Adam Wojciechowski

Poznan University of Technology, Institute of Computing Science
ul. Piotrowo 2, 60-965 Poznan, Poland
Adam.Wojciechowski@put.poznan.pl

Abstract. One of the most spectacular phenomenon that appeared over the Internet in 2005 was pixel advertisement or, to be more precise, web page with graphical micro-ads. The freshness and extreme simplicity of the concept to sell small 100-pixel squares on a virtual billboard to publish an image linking to another resource was a subject of many articles published in the Internet. However, success of MillionDollarHomePage.com project was hard to repeat. The winner took the cream. In the paper I try to identify and analyze success factors and weak points of the new concept in the Internet marketing. Several changes and improvements are proposed to focus users attention on the geographical context of micro-ads placed on a map in the background. Conceptual work is illustrated by a prototype system offering proposed features.

1 Introduction

One of the most spectacular phenomenon that appeared over the Internet in 2005 was pixel advertisement or, to be more precise, web page with graphical micro-ads. The freshness and extreme simplicity of the concept to sell small 100-pixel squares on a virtual billboard to publish an image linking to another resource was a subject of many articles published in the Internet. However, success of MillionDollarHomePage.com project is still hard to repeat. Even a quick review [9] of pixel-ad services trying to follow the origin gives us a clue that a key to focus customers attention is adding new functionality to the service. The concept itself appeared very catchy, but advertisers expect the visitors to return and interact with the information provided (e.g. read the content or click a link).

In the paper I try to identify and analyze success factors and weak points of the new concept in the Internet marketing. Several changes and improvements are proposed to focus users attention on the geographical context of micro-ads placed on a map in the background. Conceptual work is illustrated by a prototype system offering proposed features.

2 Internet as a Platform for Advertising

Internet is currently the platform for the most effective marketing of brand, products, and services [1]. There are several reasons why advertisers decide to use Internet

[*] This research was partially supported by the grant 91-429/BW.

K. Bauknecht et al. (Eds.): EC-Web 2006, LNCS 4082, pp. 232 – 241, 2006.

technology in their marketing programs. The key issue is cost-effectiveness of such campaign. Another reason is global access to information presented on websites and possibility to serve information (or sell products) in a non-stop (24/365) mode. The customers may access web resources at their time within the period required to understand given information. We must also take into account a wide variety of ways to focus customers' attention on a specific product. Mixture of text, graphics, animation, voice or sound background plus possibility of interactive contact with the customer offered in Internet marketing has no precedent in any other type of media. In fact appearing a new technique of web promotion does not eliminate the old ones but opens a new communication channel which is soon combined with other forms of communication creating new advertising service. More than that we can currently observe a renaissance of text ads which are reported to be *viewed most intently* [2]. It is relatively easy to match text advertising with the context of textual content of a document where the ad is placed. An example of such a system based on keywords matching is *AdSense* program served by Google [3]. Advertisements displayed on web pages participating in the program correspond to the context of page content which makes the delivery of the ads well targeted.

3 Huge Graphics, Small Fonts

A picture is worth a million words. Especially when the picture makes a positive association in our mind it may remain in our memory for long and make knowledge acquisition more effective. Marketers know the above truth very well and use images in advertising. What comes from *Eyetrack III* research [2] on human eye and brain perception while reading on-line content is that:

- larger on-line images hold the eye longer than smaller pictures,
- small fonts encourage focused viewing while large type promotes lighter scanning.

In this context it is quite reasonable to place huge banners on a web page or display extra large pup-up windows with graphical ads if you want to make your advertising effective. But such a practice rises objection in visitors' minds – 'I want to go directly to the content!'. They tend to use tools that disable pop-up function in their web-browsers or install programs that detect images suspected to be ads avoiding displaying them on the screen (by the way it makes the web page load faster). In reaction, advertisers change sizes of graphical ads to make automatic ads removing more difficult and the chase goes on. Will the visitors accept ads if they are smaller? How far should the graphical ad be shrank to be accepted be those who install *ad-killers*? Will the advertiser pay the same money for very small image instead of huge banner but displayed on more screens?

4 Small Graphics, No Space for Text – But It Worked… Once

Absolutely the opposite to 'huge graphics' approach to displaying graphical links to web resources was introduced by Alex Tew, a 21 year old student from UK who lunched a project named MillionDollarHomePage.com, see fig. 1. Tew made a table of million pixels divided into 10.000 blocks of 10x10 pixel in size and used to sell them for US$ 100 per square (US$ 1 per pixel). The site went live on August 26, 2005. The most surprising in the story is that Tew had a great success in selling the ‘squares’. He also had the luck to focus press interest on spectacular growth of his service. On September 22, 2005, when BBC News reported [4] about the million dollar project he had sold US$ 74,000 worth of space [5]. Within the next week he more than doubled the sales.

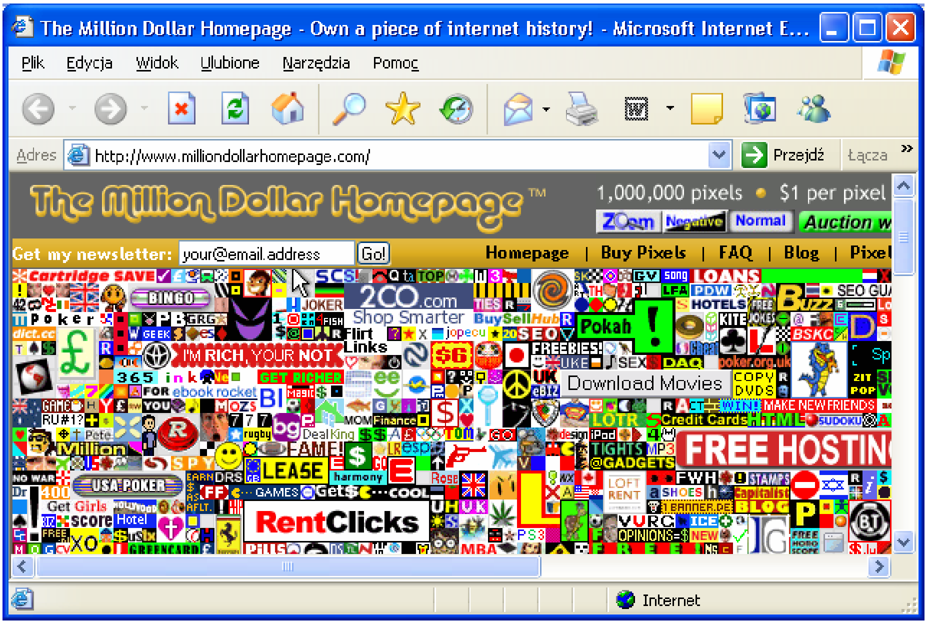

The Million Dollar Homepage - Own a piece of internet history! - Microsoft Internet E...
Plik
Edycja
Widok
Ulubione
Narzędzia
Pomoc
Adres
http://www.milliondollarhomepage.com/
Przejdź
Łącza
The Million Dollar Homepage™
1,000,000 pixels • $1 per pixel
Zoom
Negative
Normal
Auction w
Get my newsletter:
your@email.address
Go!
Homepage | Buy Pixels | FAQ | Blog | Pixel
Internet

Fig. 1. MillionDollarHomePage.com website

There is no question that the greatest popularity of MillionDollarHomePage.com came after BBC reportage. It was a real speed-up. Alex Tew earned enough to cover his studies, however he deferred the university and focused on the business. He employed a press officer in the US and quite soon 50 per cent of advertisers were coming from the US alone.

At the same time when Tew’s website was quickly growing in popularity several webmasters tried to follow the success and opened similar services (complete software packages to run a website similar to milliondollarhomepage.com are available for less than US$ 50). What appeared quite soon copy-cats had very few advertisers and Tew himself sad to Financial Times: *The idea only works once and relies on novelty* [6]. The success story goes on, although Alex Tew sold the last 1,000 pixel field on his site at eBay auction for US$ 38,100 on January 11, 2006 [7].

5 Why Visitors Do Not Return to MillionDollarHomePage.com

In the end of 2005 when I first red about the MillionDollarHomePage.com project I asked myself if I would like to make it my starting page. The answer was “No”, although I was really impressed by the colorful mixture of micro-ads. I also asked the same question my computer science students. Among 70 people there was no one who would be eager to save the address MillionDollarHomePage.com as a starting page in their web browsers. In fact, students declared that possibly they would show the web page to their friends as a phenomenon but they will not return for links nor any other knowledge. Website traffic meter confirms our observation (see Fig. 2). People do not want to return to a web page worth US$ 1m!

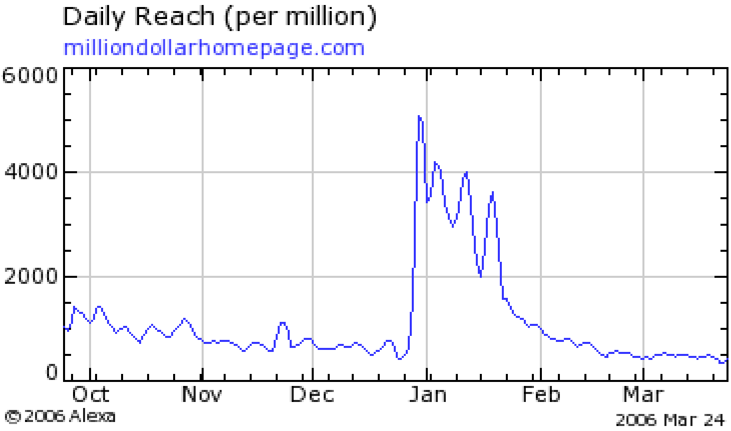


Fig. 2. Website traffic graph for milliondollarhomepage.com. Source: www.alexa.com.

5.1 What Is Missing in MillionDollarHomePage.com

Continuing discussion with my students on ‘How to improve the concept of pixel-ad web-page to make it attractive enough to return’ I organized a brain storm with a group of students. The key features proposed during the discussion were the following:

- a search tool to find or an easy-to-use tool to filter advertisers (by category, keywords, etc.),
- dynamic placement of the logos (ads) on the board in such an order to create, an image or something like an ASCII Art (by the way it is an interesting example an application of solving ‘optimal placement’ problem),
- a tool forcing frequent changes of content (originally fields on the MillionDollarHomePage.com are sold for ‘at least five years’ and although it is possible to change the image or URL we expect that this function will not be used often if it is not forced by the system).

5.2 Why People Do Not Click on Ads

While it is hard to say why other people *do* or *do not do* something we tried to find reasons why we do not like click on advertising, banners, etc. After a pretty long discussion we agreed and grouped our threats into several sets:

- missing a clear rule on 'what will happen when I click'. Action trigged after clicking on a link is defined and *known* to the web-browser (e.g. whether the new/linked document will be opened in the same or a new window). Users may only guess what may happen unless they read and understand source file of the document in the web-browser. And even after analyzing the source of the link user cannot be sure about behavior of the document downloaded to the web-browser. Opened document may have defined actions that will be run on the event of closing web-browser window,
- information about linked resource (real full address) displayed on the status line (bottom part of web-browser window) is very often hidden to the user or presented in descriptive manner (i.e. instead of 'http://www.mywebsite.com/sht.php?refid=1' the web-browser shows information 'Welcome to my site'). The text presented on the status line may be freely defined by the author of the web page,
- wide variety of file formats that are interpreted and executed on client site, which rises risk of introducing computer viruses (trojan horses, etc.) to the local system. If the real URL is hidden to the visitor s/he does not know what file is transferred to the local system and opening such resource is a 'blind date'.

We are aware of the fact that there are techniques of blocking dynamic content in web-browsers, but by disabling those dynamic functions users loose many effects that are safe for their systems.

5.3 The Objective of Advertising

The objective of advertising is to inform, to persuade, and to remind [8]. The content of MillionDollarHomePage.com is static and thus the service is awaiting for visitors. Even if we agree that there is a bit of information in the image displayed on this multi-ad-billboard and we direct the visitor via link to complete source of information, we still miss two objectives of advertising: to persuade and to remind. In fact we cannot expect that software tools would be intelligent enough to persuade the visitor to buy something thus selecting arguments and composing suggestive messages remain human domain. Finally, repeatable visits and possibility to remind advertising message requires that the system offers the visitor *fresh* information – a piece of knowledge that the visitor consider valuable at access time. Alex Tew's approach – static page of graphical links – is useless in this case, especially now, when all the fields have been sold and we cannot expect frequent changes in the content. Dynamic changes of the content seem to be the key to achieve multiple visits of customers. It is also worth to consider other bonus for visitors, e.g. free interactive gadgets (screen savers, software to download, etc.) – something that our target group may consider worth revisiting. A good example here is service PixMeUp.com. On every day they give a prize of US$100 to one of the visitors clicking on advertisements.

Size of an advertisement has influence on visitors' perception. The bigger the ad the more suggestive it is to customers [2]. But following Alex Tew's concept to place hundreds of ads on the same screen we cannot dedicate substantial part of the screen to a single graphic. There is a need to find a hybrid solution – display bigger image on request, e.g. when user moves mouse cursor above a corresponding miniature ad.

6 Improvement Proposal

Alex Tew's concept on displaying multiple graphical micro-ads on a single web-page has numerous mutations. A comprehensive review on different applications of pixel pages is available at *MDHP Watch* [9]. However majority of services analyzed by *MDHP Watch* are simple copies of the origin and, what was mentioned above, have little chance for success. Looking for success in domain of pixel ads we must consider extending functionality to provide users with extra value – meta information coming from context of advertisement placement.

6.1 Geographical Context of Graphical Advertisement

There are plenty of applications where it is recommended to focus visitor's attention on particular group of advertisements. However we know nothing about our anonymous guest and his preferences, interests etc. and thus we may present all the information we have but in such an order that the guest will find appropriate subset in natural intuitive way. One of the ways to arrange micro-ads on the screen in 'logical' order is to place them on a map. If the background of the virtual billboard is a map then placement of an image representing advertisement provides additional geographical context to the ad.

For practical reasons the map being a background must be prepared in light colors, maybe even a little blured. Juicy, intensive colors of background could dominate over content (small icons or images).

Some parts of the map are more important than the other, e.g. names of main streets on a city map should not be covered by ads to ensure that one can easily orient oneself on the map. Map-based approach is sensitive to number of ads placed on the billboard. If the map is completely covered then geographical context of the ads is lost. To avoid this risk we can propose two approaches:

- taboo list – list of fields on the virtual billboard where it is prohibited to place an ad. Taboo list of fields on the map which are not available to place an ad is defined by the system administrator,
- cover ratio – part (percent) of fields that can be occupied at a time.

In our implementation we use a hybrid solution including both above conditions.

6.2 Logical Filters Based on Keywords

In order to avoid dynamic generation of entire billboard build-up of several hundred pieces most frequently used approach assumes that the billboard is generated at server side on the event of adding or removing an ad. Let's assume a situation, where each ad is additionally described by a list of keywords. Values assigned to each keyword define subsets of advertisements sharing some properties. System administrator may define several filters – several queries to the database that return a subset of ads that satisfy specified condition (e.g. 'property = *house*'). In this case we can consider automatic generation of several virtual billboards (background map with miniature graphical ads stuck on the map at their location) on the event of adding/removing an ad to/from database. This approach is justified by speed of the application. On-line generation of a billboard made of hundreds of ads on a very busy web-server takes time that may even count in seconds.

The approach proposed above where users cannot freely specify queries but may choose from a number of views based on filters defined by administrator seem to be a good compromise. Users have the opportunity to see ads from various perspectives and system works effectively because the complex graphics are generated only on the event of change in content.

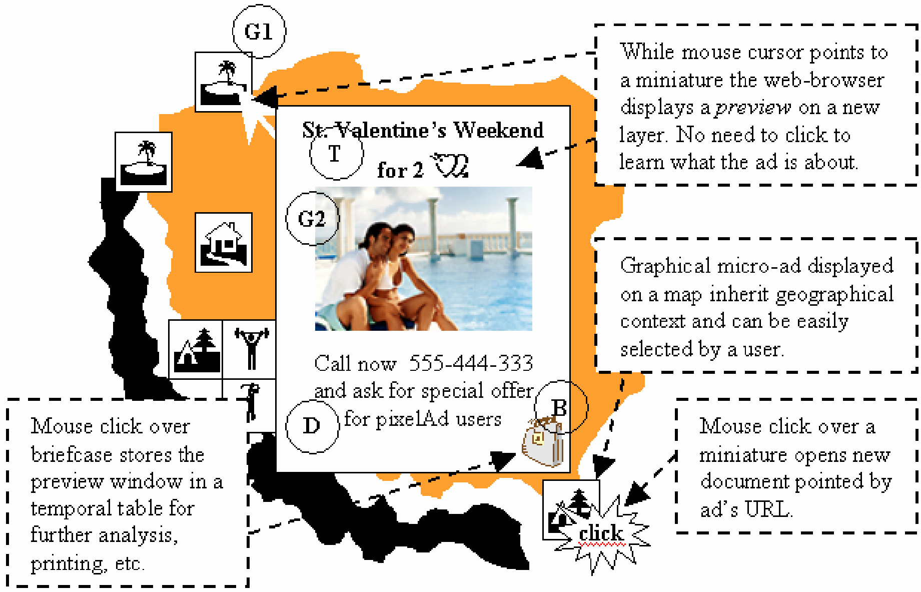
6.3 No Need to Click to See More

Being aware of the threats of visitors who are not very likely to click on links (see chapter 5.2) we can provide them with *preview*. In our case, we decided that each advertisement will be represented by a miniature graphic (G1), medium size graphic (G2), title (T), description (D) and URL, optionally definition may be extended by values of keywords. Once a visitor moves mouse cursor over a miniature graphical ad the web-browser opens a preview window on a new layer (see Fig. 3). The preview window closes automatically when mouse cursor leaves the area of miniature ad or opened preview area.

Miniature graphic still preserves the function of a link to a resource intended by the advertiser. Mouse click over a miniature image opens a new browser window and loads linked document.

6.4 Virtual Ad-Briefcase: Temporal Storage

Pixel-ad system equipped with preview function described in chapter 6.3 can serve as a repository of link (like the origin: MillionDollarHomePage.com) or to be more like a billboard with short textual or graphical messages. Viewing hundreds of advertisements displayed on one screen while only some pieces of information may be valuable to the user requires making notes. In order to collect *previews* that focused user's attention and note them with computer mouse we can propose a virtual briefcase. Once the information given in *preview* seems valuable to the user s/he can store the ad (its content) in a table named *ad-briefcase* by clicking a briefcase icon in bottom-right corner of preview window, see Fig. 3 (B). The ad-briefcase is intended to store ads selected by the user for further processing: e.g. printing or comparing on the screen. Ad-briefcase may be displayed and printed on user request. To keep the notes private for the user the ad-briefcase is cleared on closing session (closing web-browser).



While mouse cursor points to a miniature the web-browser displays a *preview* on a new layer. No need to click to learn what the ad is about.

Graphical micro-ad displayed on a map inherit geographical context and can be easily selected by a user.

Mouse click over briefcase stores the preview window in a temporal table for further analysis, printing, etc.

Mouse click over a miniature opens new document pointed by ad's URL.

Fig. 3. Schematic visualization of proposed improvements to pixel-ad system

6.5 Short Period of Advertisement Exposition

In order to keep advertisement and messages up-to-date it is necessary to force content refreshing. In our approach exposition period (the time when advertisement is displayed on the map) is a configurable parameter. However we recommend to keep it relatively short, e.g. two weeks or one month. Of course the period depends on domain of application.

Owner of an advertisement that is approaching the end of exposition is informed by email sent to him several (e.g. four) days before the end. During remaining time the exposition period may be extended but it requires the owner to manually confirm the content. If exposition period of an ad is not extended it is automatically removed from database, from the map and place occupied by the miniature ad becomes available for new advertisers.

7 Applications

New functions of pixel-ad system proposed in the paper have been implemented in a prototype system *microBan* [10]. Current and still changing prototype version of the system can be accessed at: www.microban.pixel.livent.pl. Currently we consider opening several publicly accessible information services. Two of them are mentioned below.

7.1 House to Let

In all academic cities students coming for studies from other places have the same problem – to find a room to live at a reasonable price, close to the university and they would like to see at least a photo of the house or rooms inside before they decide to go to meet the owner and negotiate the price. Pixel-ads located on a city map seem to be a good solution for those who want to let and to rent a house or a flat. Typical advertisement provided in local newspapers or real-estate offices usually miss geographical context and even if they are described by address, street name says very little to newcomers in the city, they need a map anyway. For them it is the best to choose from alternatives presented on a map.

7.2 ECDL Exams

Another application area where we plan to use proposed system is an information message board where ECDL (European Computer Driving Licence) examination centers may publish information about dates of next exams planned across the country. Quite many people willing to join the exams do manual search over web pages containing partial information on exams organized by particular examination center. Collecting this distributed knowledge in one information system with intuitive 'search tool' – the map – for manual selecting examination center near the place where the candidate lives will make it much easier to find a location to join the exams.

8 Summary

Pixel-ad systems grow in popularity since year 2005 when Alex Tew's MillionDollarHomePage.com project had a spectacular success selling US$ 1m worth of graphical micro-ads on a static web-page. Unfortunately, majority of services that try to repeat financial success of the first project do not offer any additional functions nor extensions that could make the services unique and offer new value for customers. In this paper there is a proposal of several new functions that could provide additional contextual information and force dynamic change of content on specific pixel-ad board built on a map in the background.

A prototype system implementing described functions has already been built and currently we are working on applying the concept to provide a publicly accessible high quality information systems.

Acknowledgement

I would like to thank my student, Radoslaw Drewa from Poznan University of Technology, who helped me in implementation of the system described in this paper.

References

[1] Collin S., Marketing on the Internet, Batsford, 1999.

[2] Outing S., Ruel L., The Best of Eyetrack III: What We Saw When We Looked Through Their Eyes, http://poynterextra.org/EYETRACK2004.

[3] Contextual Advertising by Google: AdSense, http://www.google.com/adsense.

[4] BBC News, Student's cash-rising net scheme, http://news.bbc.co.uk/1/hi/england/wiltshire/4271694.stm

[5] The Million Dollar Home Page Review, http://www.i4u.com/article4285.html

[6] Fontanella J., Dollar-per-pixel ad site nets student $1m, Financial Times, January 11, 2006, http://news.ft.com/cms/s/d4edefe0-82ae-11da-ac1f-0000779e2340.html

[7] Own the Last 1,000 Pixels on MillionDollarHomePage.com, http://cgi.ebay.co.uk/ws/eBayISAPI.dll?ViewItem&item=5652179487&ssPageName=ADME:L:LCA:UK:31

[8] Kotler P., Armstrong G., Principles of Marketing, Prentice Hall, 2003

[9] Million Dollar Home Page Watch, http://www.milliondollarhomepage.ws

[10] Drewa R., Analysis of selected method and tools for advertising on WWW, master thesis (in Polish), Poznan University of Technology, Poland 2006.

Author Index