Il-Yeol Song et al. (Eds.)

LNCS 5232

# Advances in Conceptual Modeling – Challenges and Opportunities

ER 2008 Workshops CMLSA, ECDM, FP-UML, M2AS, RIGiM, SeCoGIS, WISM
Barcelona, Spain, October 2008, Proceedings

Springer

# Lecture Notes in Computer Science 5232

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

Il-Yeol Song et al. (Eds.)

# Advances in Conceptual Modeling – Challenges and Opportunities

ER 2008 Workshops CMLSA, ECDM, FP-UML, M2AS, RIGiM, SeCoGIS, WISM
Barcelona, Spain, October 20-23, 2008
Proceedings

Springer

Volume Editors

Il-Yeol Song, E-mail: songiy@drexel.edu
Mario Piattini, E-mail: mario.piattini@uclm.es
Yi-Ping Phoebe Chen, E-mail: phoebe@deakin.edu.au
Sven Hartmann, E-mail: sven.hartmann@tu-clausthal.de
Fabio Grandi, E-mail: fabio.grandi@unibo.it
Juan Trujillo, E-mail: jtrujillo@dlsi.ua.es
Andreas L. Opdahl, E-mail: andreas.opdahl@uib.no
Fernando Ferri, E-mail: fernando.ferri@irpps.cnr.it
Patrizia Grifoni, E-mail: patrizia.grifoni@irpps.cnr.it
Maria Chiara Caschera, E-mail: mc.caschera@irpps.cnr.it
Colette Rolland, E-mail: colette.rolland@univ-paris1.fr
Carson Woo, E-mail: carson.woo@ubc.ca
Camille Salinesi, E-mail: camille.salinesi@univ-paris1.fr
Esteban Zimányi, E-mail: ezimanyi@ulb.ac.be
Christophe Claramunt, E-mail: christophe.claramunt@ecole-navale.fr
Flavius Frasincar, E-mail: frasincar@few.eur.nl
Geert-Jan Houben, E-mail: geert-jan.houben@vub.ac.be
Philippe Thiran, E-mail: philippe.thiran@fundp.ac.be

Library of Congress Control Number: 2008935898

CR Subject Classification (1998): H.2-5, F.4.1, D.2, C.2.4, I.2, J.1, J.3

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

ISSN 0302-9743
ISBN-10 3-540-87990-0 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-87990-9 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: 12533329 06/3180 5 4 3 2 1 0

# Preface

We would like to welcome you to the proceedings of the workshops held in conjunction with the 27th International Conference on Conceptual Modeling (ER 2008). While the ER main conference covers a wide spectrum of conceptual modeling research, increasingly complex real-world problems demand new perspectives and active research in new applications. The ER workshops attempt to provide researchers, students, and industry professionals with a forum to present and discuss emerging hot topics related to conceptual modeling.

We received 13 excellent proposals for workshops to be held with ER 2008. We accepted the following seven based on peer reviews:

1. The Second International Workshop on Conceptual Modeling for Life Sciences Applications (CMLSA 2008), organized by Yi-Ping Phoebe Chen and Sven Hartmann.
2. The 5th International Workshop on Evolution and Change in Data Management (ECDM 2008), organized by Fabio Grandi.
3. The 4th International Workshop on Foundations and Practices of UML (FP-UML 2008), organized by Juan Trujillo and Andreas L. Opdahl.
4. The First International Workshop on Modeling Mobile Applications and Services (M2AS 2008), organized by Fernando Ferri, Patrizia Grifoni, and Maria Chiara Caschera.
5. The Second International Workshop on Requirements, Intentions and Goals in Conceptual Modeling (RIGiM 2008), organized by Colette Rolland, Carson Woo, and Camille Salinesi.
6. The Second International Workshop on Semantic and Conceptual Issues in Geographic Information Systems (SeCoGIS 2008), organized by Esteban Zimányi and Christophe Claramunt.
7. The 5th International Workshop on Web Information Systems Modeling (WISM 2008), organized by Flavius Frasincar, Geert-Jan Houben, and Philippe Thiran.

These seven workshops received 18, 8, 12, 23, 15, 20, and 12 papers, respectively. Following the rule of the ER workshops, the respective workshop Program Committee carried out peer reviews and accepted 6, 3, 5, 9, 6, 9, and 5 papers, with acceptance rates of 33%, 38%, 42%, 39%, 45%, 40%, and 42%, respectively. In total, 108 workshop papers were received and 42 papers were accepted with the average acceptance rate of 39%.

We also had invited speakers and papers that significantly enhanced the perspectives and quality of the ER 2008 workshops. The four invited papers are:

1. CMLSA 2008: Victor Maojo, "Ontologies in Practice: From Biomedical Informatics to Nanomedicine."
2. ECDM 2008: Carlo Zaniolo, "Time Versus Standards: A Tale of Temporal Databases."

3. FP-UML 2008: Yair Wand, “Using Object Concepts and UML for Conceptual Modeling.”
4. WISM 2008: Hui Ma, Klaus-Dieter Schewe, Bernhard Thalheim, and Qing Wang, “Abstract State Services—A Theory of Web Services.”

We thank the Organizing Chairs of the seven workshops. They invested an enormous amount of time and effort for the workshops, handling the paper submissions, organizing Program Committees, reviewing, selecting workshop papers, and collecting camera-ready copies for the workshops. Most of all, we would like to express our sincere appreciation to the authors who contributed their hard works as well as to the members of the Program Committees and external reviewers who ensured high-quality programs, resulting in this outstanding program. We are also indebted to the ER 2008 Organization Committee for their support in reviewing workshop proposals and scheduling workshops.

We hope all participants shared the recent advances in the emerging areas and found some opportunities and challenges in these new research areas in conceptual modeling.

October 2008

Il-Yeol Song
Mario Piattini

# Organization

## ER 2008 Workshop Chairs

| | |
|---|---|
| Il-Yeol Song | Drexel University, Philadelphia, USA |
| Mario Piattini | Universidad de Castilla-La Mancha, Ciudad Real, Spain |

## CMLSA 2008 Program Chairs

| | |
|---|---|
| Yi-Ping Phoebe Chen | Deakin University, Australia |
| Sven Hartmann | Clausthal University of Technology, Germany |

## CMLSA 2008 Publicity Chair

| | |
|---|---|
| Markus Kirchberg | Institute for Infocomm Research, A*STAR, Singapore |

## CMLSA 2008 Program Committee

| | |
|---|---|
| Jake Chen | Indiana University, USA |
| Carlo Combi | University of Verona, Italy |
| Silke Eckstein | Braunschweig University of Technology, Germany |
| Amarnath Gupta | University of California San Diego, USA |
| Dirk Labudde | Dresden University of Technology, Germany |
| Dirk Langemann | University of Lübeck, Germany |
| Sebastian Link | Victoria University of Wellington, New Zealand |
| Huiqing Liu | BioMaPS, Rutgers University, USA |
| Victor Maojo | Polytechnic University of Madrid, Spain |
| Maria Mirto | University of Salento, Italy |
| Roque Marín Morales | University of Murcia, Spain |
| Fabio Porto | EPF Lausanne, Switzerland |
| Sudha Ram | University of Arizona, USA |
| Allan Rodrigo | University of Auckland, New Zealand |
| Keun Ho Ryu | Chungbuk National University, Korea |
| Amandeep S. Sidhu | Curtin University of Technology, Australia |
| Robert Stevens | University of Manchester, UK |
| Marie-Noëlle Terrasse | University of Burgundy, France |

| | |
|---|---|
| Thodoros Topaloglou | University of Toronto, Canada |
| Jing Wang | Massey University, New Zealand |
| Xiaofang Zhou | University of Queensland, Australia |
| Esteban Zimányi | Université Libre de Bruxelles, Belgium |

## CMLSA 2008 External Referees

Geoffroy Cruxifix
Gabriel Fung
José M. Juarez
Andreas Kupfer
José Palma
Sudipto Saha

## ECDM 2008 Workshop Chair

| | |
|---|---|
| Fabio Grandi | University of Bologna, Italy |

## ECDM 2008 Program Committee

| | |
|---|---|
| Alessandro Artale | Free University of Bolzano-Bozen, Italy |
| Sourav Bhowmick | Nanyang Technical University, Singapore |
| Mike Boehlen | Free University of Bolzano-Bozen, Italy |
| Carlo Combi | University of Verona, Italy |
| Curtis Dyreson | Utah State University, USA |
| Shashi Gadia | Iowa State University, USA |
| Michel Klein | Vrije Universiteit Amsterdam, The Netherlands |
| Federica Mandreoli | University of Modena and Reggio Emilia, Italy |
| Richard McClatchey | University of the West of England, UK |
| Torben Bach Pedersen | Aalborg University, Denmark |
| Erik Proper | Radboud Universiteit Nijmegen, The Netherlands |
| Sudha Ram | University of Arizona, USA |
| John Roddick | Flinders University, Australia |
| Myra Spiliopoulou | Otto-von-Guericke-Universität Magdeburg, Germany |
| Carlo Zaniolo | UCLA, USA |

## ECDM 2008 External Referees

Marijke Keet
Riccardo Martoglia
Hyun Moon

## FP-UML 2008 Workshop Steering Committee

| | |
|---|---|
| Juan Trujillo | University of Alicante, Spain |
| Il-Yeol Song | Drexel University, USA |
| Jeffrey Parsons | Memorial University of Newfoundland, Canada |

## FP-UML 2008 Program Co-chairs

| | |
|---|---|
| Juan Trujillo | University of Alicante, Spain |
| Andreas L. Opdahl | University of Bergen, Norway |

## FP-UML 2008 Program Committee

| | |
|---|---|
| Doo-Hwan Bae | KAIST, South Korea |
| Michael Blaha | OMT Associates Inc., USA |
| Cristina Cachero | University of Alicante, Spain |
| Tharam Dillon | University of Technology Sydney, Australia |
| Brian Dobing | University of Lethbridge, Canada |
| Dirk Draheim | Freie Universität Berlin, Germany |
| Joerg Evermann | Victoria University, Wellington, New Zealand |
| Eduardo Fernández | University of Castilla-La Mancha, Spain |
| Briand Henderson-Sellers | University of Technology, Sydney, Australia |
| Miguel Katrib | University of La Habana, Cuba |
| Jens Lechtenbörger | Universität Münster, Germany |
| Tok Wang Ling | National University of Singapore, Singapore |
| Pericles Loucopoulos | University of Manchester, UK |
| Hui Ma | Massey University, New Zealand |
| Heinrich C. Mayr | University of Klagenfurt, Austria |
| Jeffrey Parsons | Memorial University of Newfoundland, Canada |
| Oscar Pastor | Technical University of Valencia, Spain |
| Witold Pedrycz | University of Alberta, Canada |
| Mario Piattini | University of Castilla-La Mancha, Spain |
| Ivan Porres | Åbo Akademi University, Finland |
| Colette Rolland | Universit Paris 1-Panthéon Sorbonne, France |
| Matti Rossi | Helsingin kauppakorkeakoulu, Finland |
| Manuel Serrano | University of Castilla-La Mancha, Spain |
| Keng Siau | University of Nebraska-Lincoln, USA |
| Il-Yeol Song | Drexel University, USA |
| Ambrosio Toval | Universidad de Murcia, Spain |
| Panos Vassiliadis | University of Ioannina, Greece |

## FP-UML 2008 External Referees

| | |
|---|---|
| Ornsiri Thonggoom | Drexel University, USA |
| Fernando Molina | Universidad de Murcia, Spain |

## M2AS 2008 Organization Committee

| | |
|---|---|
| Fernando Ferri | IRPPS-CNR, Rome, Italy |
| Patrizia Grifoni | IRPPS-CNR, Rome, Italy |
| Maria Chiara Caschera | IRPPS-CNR, Rome, Italy |

## M2AS 2008 Program Committee

| | |
|---|---|
| Frederic Andres | National Institute of Informatics, Japan |
| Paris Avgeriou | University of Groningen, The Netherlands |
| Regina Bernhaupt | University of Salzburg, Austria |
| Michela Bertolotto | University College Dublin, Ireland |
| Antonio Fernández-Caballero | University of Castilla-La Mancha, Spain |
| Augusto Celentano | University of Venice, Italy |
| Richard Chbeir | Bourgogne University, France |
| Fang Chen | National ICT, Australia |
| Deborah Dahl | Conversational Technologies, USA |
| Alfred Dielmann CSTR | University of Edinburgh, UK |
| Chirine Ghedira | Claude Bernard Lyon I University, France |
| Ho-fung Leung | Chinese University of Hong Kong, China |
| Oscar Pastor López | Technical University of Valencia, Spain |
| Paolo Merialdo | University of Rome 3, Italy |
| Amit Anil Nanavati | IBM Research, India |
| Nitendra Rajput | IBM Research, India |
| Thomas Strang | German Aerospace Center (DLR), Germany |
| David Taniar | Monash University, Australia |
| Riccardo Torlone | University of Rome 3, Italy |
| Olga De Troyer | WISE, Belgium |
| Iñaki Vázquez | University of Deusto, Spain |

## RIGiM 2008 Workshop Organizers

| | |
|---|---|
| Colette Rolland | Université Paris 1 Panthéon Sorbonne, France |
| Carson Woo | University of British Columbia, Canada |
| Camille Salinesi | Université Paris 1 Panthéon Sorbonne, France |

## RIGiM 2008 Program Committee

| | |
|---|---|
| Ian Alexander | Scenario Plus, UK |
| Daniel Amyot | University of Ottawa, Canada |
| Mikio Aoyoma | Nanzan University, Japan |
| Aybuke Aurum | University of New South Wales, Australia |
| Franck Barbier | University of Pau, France |
| Daniel Berry | University of Waterloo, Canada |

| | |
|---|---|
| Lawrence Chung | University of Texas at Dallas, USA |
| Luiz Cysneiros | York University, Canada |
| Eric Dubois | Centre de Recherche Public Henri Tudor, Luxembourg |
| Vincenzo Gervasi | University of Pisa, Italy |
| Aditya K. Ghose | University of Wollongong, Australia |
| Jaap Gordijn | Vrije Universiteit Amsterdam, The Netherlands |
| Peter Haumer | IBM Rational, USA |
| Aneesh Krishna | University of Wollongong, Australia |
| John Krogstie | Norwegian University of Science and Technology, Norway |
| Lin Liu | Tsinghua University, China |
| Peri Loucopoulos | Loughborough University, UK |
| John Mylopoulos | University of Toronto, Canada |
| Selmin Nurcan | Université Paris 1 Panthéon Sorbonne, France |
| Andreas Opdahl | University of Bergen, Norway |
| Barbara Pernici | Politecnico di Milan, Italy |
| Yves Pigneur | University of Lausanne, Switzerland |
| Klaus Pohl | University of Duisburg-Essen, Germany |
| Jolita Ralyte | University of Geneva, Switzerland |
| Bjorn Regnell | Lund University, Sweden |
| Motoshi Saeki | Tokyo Institute Of Technology, Japan |
| Pnina Soffer | University of Haifa, Israel |
| Carine Souveyet | Université Paris 1, France |
| Yair Wand | University of British Columbia, Canada |
| Roel Wieringa | University of Twente, The Netherlands |

## SeCoGIS 2008 Workshop Chairs

| | |
|---|---|
| Esteban Zimányi | Université Libre de Bruxelles, Belgium |
| Christophe Claramunt | Naval Academy Research Institute, France |

## SeCoGIS 2008 Steering Committee

| | |
|---|---|
| Claudia Bauzer Medeiros | University of Campinas, Brazil |
| Michela Bertolotto | University College Dublin, Ireland |
| Jean Brodeur | Natural Resources Canada |
| Christophe Claramunt | Naval Academy Research Institute, France |
| Christelle Vangenot | Ecole Polytechnique Fédérale de Lausanne, Switzerland |
| Esteban Zimányi | Université Libre de Bruxelles, Belgium |

## SeCoGIS 2008 Program Committee

| | |
|---|---|
| Alia I. Abdelmoty | Cardiff University, UK |
| Gennady Andrienko | Fraunhofer Institute AIS, Germany |
| Natalia Andrienko | Fraunhofer Institute AIS, Germany |
| Yvan Bédard | Université de Laval, Canada |
| Michela Bertolotto | University College Dublin, Ireland |
| James D. Carswell | Dublin Institute of Technology, Ireland |
| Eliseo Clementini | University of L'Aquila, Italy |
| Maria Luisa Damiani | University of Milan, Italy |
| Clodoveu Davis | Pontificia Universidade Catolica de Minas Gerais, Brazil |
| Max Egenhofer | NCGIA, USA |
| Fernando Ferri | Istituto di Ricerche sulla Popolazione e le Politiche Sociali, Italy |
| Andrew Frank | Technical University of Vienna, Austria |
| Anders Friis-Christensen | European Commission Joint Research Centre, Italy |
| Antony Galton | University of Exeter, UK |
| Bo Huang | University of Calgary, Canada |
| Ki-Joune Li | Pusan National University, South Korea |
| Thérèse Libourel | Université de Montpellier II, France |
| Miguel R. Luaces | University of Coruna, Spain |
| Jose Macedo | Ecole Polytechnique Fédérale de Lausanne, Switzerland |
| Pedro Rafael Muro Medrano | Universidad de Zaragoza, Spain |
| Peter van Oosterom | Delft University of Technology, The Netherlands |
| Dimitris Papadias | Hong Kong University of Science and Technology, Hong Kong |
| Dieter Pfoser | CTI, Greece |
| Ricardo Rodrigues Ciferri | Universidade Federal de São Carlos, Brazil |
| Andrea Rodriguez | University of Concepcion, Chile |
| Sylvie Servigne-Martin | INSA de Lyon, France |
| Stefano Spaccapietra | Ecole Polytechnique Fédérale de Lausanne, Switzerland |
| Emmanuel Stefanakis | Harokopio University of Athens, Greece |
| Kathleen Stewart Hornsby | University of Iowa, USA |
| Christelle Vangenot | Ecole Polytechnique Fédérale de Lausanne, Switzerland |
| Antonio Miguel Vieira Monteiro | Instituto National de Pesquisas Espaciais, Brazil |
| Nancy Wiegand | University of Wisconsin-Madison, USA |

## WISM 2008 Workshop Co-chairs

| | |
|---|---|
| Flavius Frasincar | Erasmus University Rotterdam, The Netherlands |
| Geert-Jan Houben | Vrije Universiteit Brussel and TU Eindhoven, Belgium and The Netherlands |
| Philippe Thiran | Namur University, Belgium |

## WISM 2008 Program Committee

| | |
|---|---|
| Bernhard Thalheim | Christian Albrechts University Kiel, Germany |
| Christopher Thomas | Wright State University, USA |
| Dimitris Plexousakis | University of Crete, Greece |
| Djamal Benslimane | University of Lyon 1, France |
| Flavius Frasincar | Erasmus University Rotterdam, The Netherlands |
| Geert-Jan Houben | Vrije Universiteit Brussel and TU Eindhoven, Belgium and The Netherlands |
| Hyoil Han | Drexel University, USA |
| Ivan Jelinek | Czech Technical University in Prague, Czech Republic |
| Jaime Gómez | Universidad de Alicante, Spain |
| Lorna Uden | Staffordshire University, UK |
| Martin Dzbor | The Open University, UK |
| Martin Gaedke | Chemnitz University of Technology, Germany |
| Michael Mrissa | Namur University, Belgium |
| Moira Norrie | ETH Zurich, Switzerland |
| Oscar Pastor | Valencia University of Technology, Spain |
| Philippe Thiran | Namur University, Belgium |
| Riccardo Torlone | Universita di Roma Tre, Italy |
| Tommaso Di Noia | Technical University of Bari, Italy |
| Zakaria Maamar | Zayed University, UAE |

# Table of Contents

## Adaptive Services and Interaction for Mobile Devices



## Mobile Systems and Architecture



## Usability, Users' Study and Application on Mobile Devices



## RIGiM 2008 – Second International Workshop on Requirements, Intentions and Goals in Conceptual Modeling

## Modeling



## Elicitation Issues



## SeCoGIS 2008 – Second International Workshop on Semantic and Conceptual Issues in Geographic Information Systems



## Foundational Aspects



## Ontologies and Location-Based Services

## Interoperability and Spatial Infrastructures



## WISM 2008 – Fifth International Workshop on Web Information Systems Modeling



## Web Information Systems



## Semantic Web Information Systems

# Preface to CMLSA 2008

Yi-Ping Phoebe Chen[1], Sven Hartmann[2], and Markus Kirchberg[3]

[1] Deakin University, Australia
[2] Clausthal University of Technology, Germany
[3] Institute for Infocomm Research, A*STAR, Singapore

Life Sciences applications typically involve large volumes of data of various kinds and a multiplicity of software tools for managing, analysing and interpreting them. There are many challenging problems in the processing of life sciences data that require effective support by novel theories, methods and technologies. Conceptual modelling is the key for developing high-performance information systems that put these theories, methods and technologies into practice. The fast growing interest in life sciences applications calls for special attention on resource integration and collaborative efforts in information systems development.

This volume contains the papers presented at the Second International Workshop on Conceptual Modelling for Life Sciences Applications (CMLSA 2008) which was held in Barcelona, Spain from October 20 to 23, 2008 in conjunction with the Twenty-Seventh International Conference on Conceptual Modeling (ER 2008). On behalf of the programme committee we commend these papers to you and hope you find them useful.

The primary objective of the workshop is to share research experiences in conceptual modelling for applications in life sciences and to identify new issues and directions for future research in relevant areas, including bioinformatics, health informatics, medical and veterinary informatics. The workshop invited original papers exploring the usage of conceptual modelling ideas and techniques for developing and improving life sciences databases and information systems. Following the call for papers which yielded 18 submissions, there was a rigorous refereeing process that saw each paper refereed by three international experts. The six papers judged best by the programme committee were accepted and are included in this volume.

We are grateful to Professor Victor Maojo from the Universidad Politécnica de Madrid who kindly agreed to present the CMLSA keynote address on *Ontologies in Practice: From Biomedical Informatics to Nanomedicine*.

We wish to thank all authors who submitted papers and all workshop participants for the fruitful discussions. We also like to thank the members of the programme committee for their timely expertise in carefully reviewing the submissions. Finally, we wish to express our appreciation to the local organisers at the Universitat Politècnica de Catalunya and the Universitat Oberta de Catalunya for the wonderful days in Barcelona.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, p. 1, 2008.

# Models of the Human Metabolism

Dirk Langemann[1] and Achim Peters[2]

[1] University of Lübeck, Institute of Mathematics
Wallstr. 40, 23560 Lübeck, Germany
langeman@math.uni-luebeck.de
[2] University of Lübeck, Medical Clinic I
Ratzeburger Allee 160, 23538 Lübeck
achim.peters@uni-luebeck.de
http://www.math.uni-luebeck.de/mitarbeiter/langeman

**Abstract.** The systemic investigation of the energy metabolism in the frame of the selfish-brain theory focusses on supply chains. They describe the transport chain beginning with the energy exploration in the remote environment until the energy is consumed by neurons in the brain.

Modeling this, the main difficulty consists in the selection of significant core models among the enormous number of known substances and regulatory mechanisms, which are afflicted with considerable uncertainties in general. We supplement the standard bottom-up modeling of certain mechanisms by a deductive approach. Therefore, we investigate general supply chains and deduce indispensable elements in the regulatory mechanisms from available observations.

A critical selection of system properties and underlying mechanisms enables us to simulate observations, which cannot be explained by the classical glucostatic and lipostatic theory. These observations are the e. g. nearly constant energy level in the brain, the different responses of the periphery and the brain to atrophic periods and finally the development of diabetes.

**Keywords:** human metabolism, supply chains, model reduction.

## 1 Introduction

In [1], a systemic approach in the analysis of the human metabolism is given. Obesity is explained as a mistuning of the energy supply chain from the resources in the environment into the final ATP delivery into the brain [2]. The approach results in an explanation of the human metabolism as a regulatory system. It includes the brain as an important energy consumer and as the administrative instance. Therefore, the brain has a very strong position in the competition for energy within the body. That is the reason why this approach is called selfish-brain theory. The selfish-brain theory [1,2,3] can explain certain – partially very old – observations, which could not be modeled by traditional approaches.

These are two elderly theories which are still rivaling nowadays. These are the glucostatic theory [4], which assumes that the blood glucose is the hierarchically highest regulated quantity in the metabolism, and the lipostatic theory [5],

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 2–11, 2008.

which presumes that the fat compartment is the highest one in the regulatory hierarchy. The second theory has become strengthened by the detection of the hormone leptin [6]. They have the assumption in common that the energy is passively distributed in the whole body. Both theories cannot explain the development of diabetes and related experimental results [7], or the observations of the asymmetric energy distribution in an atrophic situation [8], of the constant energy level in the brain [9], which is nearly independent of exogenous influences, or of the occurrence of appetite when the stores of blood glucose or fat, respectively, are filled. Another fact, which cannot be included in these theories, is the supply of the neurons by energy on demand [10], what indicates the existence of the strong pulling mechanism regulating the ATP level in the brain.

The use of regulatory systems in the selfish-brain theory involves some questions about modeling in medicine and life science in general, because most of the relations are not as quantified as in physics or engineering. Modeling in physics follows a bottom-up approach in general. Detailed mechanisms, which are quantitatively known, are combined in a model, which enables us to simulate more complex phenomena and to understand the interaction of different mechanisms. In contrast to that, we are faced to an enormous number of mechanisms in medicine and life science, and most of them are not yet quantified [11]. It is extremely difficult to create experiments which separate a single mechanism from all other influences. This is impressively demonstrated in the identification of the network structure of numerous subsystems of the metabolism in [12]. On the other hand, regulatory mechanisms are the product of an evolution and are assumed to be stable with respect to disturbances. These points motivate a top-down approach, where simple models reproducing fundamental observations are refined step by step. This has been successfully done for different problems in [13,14], where e. g. indispensable components of the synaptic plasticity and of the appetite regulation, respectively, have been identified.

The present paper is based on the mathematical investigation of general supply chains and the different roles of push and pull components in [3] and on a deductive assignment of elements in the short-term appetite regulation in [14]. There, a number of phenomena concerning the appetite is modeled by a two-compartment model with a non-linear appetite activation. A sketch in [3] extends this very simple model to a short-term model including energy containing compartments and signals like the activation of the ventromedial hypothalamus (VMH) governing the energy allocation and the lateral hypothalamus (LH) governing the appetite [15,16,17]. Here, we firstly present and analyze a resulting long-term model integrating over the daily cyclic behavior.

The long-term model is given in Sec. 2. Due to its small size of only four compartments, it can be completely analyzed by mathematical methods in Sec. 3. We see that fundamental properties are independent of the parameter choice, e. g., there is exactly one stable stationary point with an unbounded attraction domain, and the size of the fat compartment increases with a weakened allocation. Three selected scenarios are modeled in Sec. 4. These are a cyclic ingestive behavior, an atrophic situation and an allocation failure. The simulations

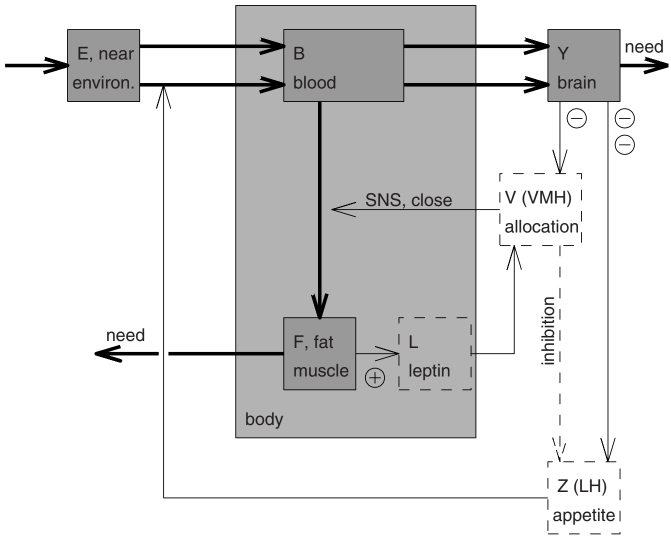



**Fig. 1.** Long-term model of the human metabolism. The near environment is abundantly filled. Push and pull mechanisms transport energy from the environment via the blood into the brain. Fat and muscles form a side compartment, which is regulated by leptin and the sympathetic nervous system governed by the ventromedial hypothalamus (VMH). The only additional regulation in the model is the blocking of the appetite in the lateral hypothalamus (LH) by an activated allocation in the VMH.

demonstrate that the presented model is able to explain the above mentioned phenomena which are not covered by the traditional approaches. Thereby, it is the first mathematical model of the human metabolism, which reproduces the observations in [7,9,8,10]. The paper finishes with a short conclusion.

The long-term model is constructed by the easiest possible relations, i. e., linear dependencies and general parameters. It is shown that the fundamental properties of the proposed model are conserved independent from the particular parameter choice. It shows some central ideas of the selfish-brain theory, and it describes the human metabolism as regulatory system with a dominant energy pull into the brain. The brain is supplied on the demand, and its energy level stays in a small constant range unaffected of exogenous influences.

Our long-term model in the frame of the selfish-brain theory explains obesity as a traffic jam in the energy supply chain from the environment into the brain. An allocation failure leads to a constrained energy transport into the brain, which as the administrative instance enforces the ingestion. The person is becoming obese.

This statement contradicts the widespread assumption that obese people simply intake too much due to a lack of self-discipline. But, let us mention that the regulation of food intake is very strict in a medium time scale. A healthy person intakes about 2500 kcal every day. Without self-reflection, most of the people conserve their weight for years. On the other hand, an additional daily tea-spoon of butter would probably lead to a weight gain of at least 2 kg per year. Thus, we can assume a strong regulation of the ingested amount of food in a medium or long time scale. Of course, a person becoming obese intakes an increased amount of energy, since energy conservation holds in a metabolic system, too. The selfish-brain theory asks why the intake is increased. In forthcoming investigations, the presented model is combined with stress models [18], metabolic learning [19] and their interaction.

## 2 Long-Term Model

Here, we form a long-term model, which is based on the simulation results of a short-time model [3] including the daily cycling in the need of the periphery describing the rhythms of sleep and alertness and of working and relaxing. The short-time model contains regulatory mechanism in the VMH and in the LH by saturated activation functions. Within the daily cycling, the activation functions periodically reach their maximum and minimum. The short-time model needs the five energy containing compartments: near environment, blood, brain, fat, muscles, and three signals, which are the insulin production, and the activations of VMH and LH. It reproduces a number of experimental observation, e. g. the effect that fast ingested glucose leads to an enforced insulin production and thus, an enforced energy transport into the fat [20,21]. Finally, the brain does not receive the required energy and generates appetite in the LH. Such a detailed model restricts the possibilities of a mathematical analysis and of numerical long-term simulations. That is the reason why it has been reduced to its long-term effects.

The long-term model presented here, reduces the number of compartments to four: the near environment, the blood glucose, the brain and the periphery compartment combining fat and muscles. The energy levels are denoted by $E$ for the environment, by $B$ for the blood glucose, by $Y$ for the brain energy level and by $F$ for the fat. The regulatory mechanism are described by linear functions, because the long-term changes are close to the equilibrium point. Furthermore, our model should not depend on the particular choice of the non-linearities, and the mechanisms are the stable result of an evolutionary development and do not depend too sensitively on exogenous influences. We have checked that physiological non-linearities do not affect the fundamental behavior of the model.

The model equations connect the time-derivatives of the energy levels in the compartments with the fluxes. That means

$$\dot{E} = j_{\mathrm{in}} - j_{\mathrm{EB}}, \tag{1}$$
$$\dot{B} = j_{\mathrm{EB}} - j_{\mathrm{BY}} - j_{\mathrm{BF}}, \tag{2}$$
$$\dot{Y} = j_{\mathrm{BY}} - j_{\mathrm{out}}^{\mathrm{Y}}, \tag{3}$$
$$\dot{F} = j_{\mathrm{BF}} - j_{\mathrm{out}}^{\mathrm{F}}, \tag{4}$$

where $j_{\mathrm{EB}}$ denotes the energy flux from the environment into the blood, shortly called ingestion, $j_{\mathrm{BY}}$ is the energy flux from the blood into the brain and $j_{\mathrm{BF}}$ is the flux from the blood into the fat. Furthermore, the system (1–4) contains the need of the brain described by an energy outflow $j^Y_{\rm out}$ and the need of the periphery $j^F_{\rm out}$. Here, the needs are assumed to be constant. The inflow into the near environment is modeled by $j_{\rm in}$. For simplicity, we fix the set-points of the energy levels in the compartments to 0, and the constitutive equations read as

$$j_{\rm EB} = k_1 E - \ell_1 B + \alpha Y, \tag{5}$$
$$j_{\rm BY} = k_2 B - \ell_2 Y, \tag{6}$$
$$j_{\rm BF} = j_{\rm fat} - \mu F + \alpha Y, \tag{7}$$
$$j_{\rm in} = -\ell_{\rm in} E. \tag{8}$$

The fluxes $j_{\rm EB}$ and $j_{\rm BY}$ are composed by a push component with the parameter $k_i > 0$ and a pull component with the parameter $\ell_i > 0$, $i \in \{1,2\}$ [10,22]. The influx $j_{\rm BF}$ from the blood into the fat is modeled by a constant term $j_{\rm fat} > 0$, which is modulated by the leptin response $\mu F$, $\mu > 0$ [23] and the allocation $\alpha Y$, $\alpha > 0$. That means a decreasing energy level in the brain diminishes the flow from the blood glucose into the fat [1,2] by activating the VMH and thus the sympathetic nervous system. This mechanism contains the diminishing of insulin. At the same time, the ingestion is blocked in Eq. (5) by a similar term [24]. Finally Eq. (8) models an abundant offer of energy in the remote environment. Until now, the only presumed restriction of the parameters is their positivity, i. e. nothing more than the physiological direction of the effects is fixed.

Hence, the equations (1–8) form a dynamical system with only four compartments and linear relations at the right-hand side. The four differential equations are coupled, and hence, the equilibrium point of the system does not coincide with the set-points of the single compartments.

## 3 Model Analysis

First, we investigate the stationary points of the dynamical system in dependency of the allocation parameter $\alpha$. The stationarity condition $\dot{E} = \dot{B} = \dot{Y} = \dot{F} = 0$ leads to a $4 \times 4$–system of linear equations in $E$, $Y$, $B$ and $F$. Its solution is unique whenever $\alpha k_2 \neq \ell_1 \ell_2$. In particular the stationary solution is

$$F_{\rm stat}(\alpha) = \frac{p\alpha + q}{r\alpha + s} \quad \text{with} \quad F'_{\rm stat}(\alpha) = \frac{ps - qr}{(r\alpha + s)^2}. \tag{9}$$

The coefficients in Eq. (9) are

$$p = k_1 k_2 (j^{\rm F}_{\rm out} + j^{\rm Y}_{\rm out}) + j_{\rm fat} k_2 \ell_{\rm in} + j^{\rm F}_{\rm out} \ell_{\rm in} (k_2 + \ell_1),$$

and

$$q = \ell_1 \ell_2 \ell_{\rm in} (j^{\rm F}_{\rm out} - j_{\rm fat})\,, \; r = k_2 \ell_{\rm in} \mu \; \text{ and } \; s = \ell_1 \ell_2 \ell_{\rm in} \mu.$$

Then, we find that the nominator of the derivative $F'_{\rm stat}(\alpha)$ fulfills

$$ps - rq = -\mu \ell_1 \ell_2 \ell_{\rm in} [(j^{\rm F}_{\rm out} + j^{\rm Y}_{\rm out}) k_1 k_2 + (2k_2 + \ell_1)\ell_{\rm in}] < 0,$$

which is negative for all choices of positive parameters. Hence, the stationary size of the fat compartment increases with a decreasing allocation strength $\alpha$ independent from the particular parameters. A similar calculation shows that the stationary blood glucose is increasing with a decreasing allocation parameter $\alpha$. It can be argued that $\ell_1\ell_2 - \alpha k_2 > 0$ or $r\alpha + s > 0$, respectively, is the physiological range of the solution because even a vanishing $\alpha$ does not affect an infinite body weight.

The stationary solution is globally stable. The Jacobian of the right hand side is

$$J = \begin{pmatrix} -k_1 - \ell_{\text{in}} & \ell_1 & -\alpha & 0 \\ k_1 & -k_2 - \ell_1 & \ell_2 & \mu \\ 0 & k_2 & -\ell_2 & 0 \\ 0 & 0 & \alpha & -\mu \end{pmatrix},$$

which is negatively definite because all minors $M_1, \ldots, M_4$ of $-J$ have positive determinants. i.e. $M_1 = k_1 + \ell_{\text{in}} > 0$, $\det M_2 = k_1k_2 + (k_2 + \ell_1)\ell_{\text{in}} > 0$, and

$$M_1 = k_1 + \ell_{\text{in}} > 0\,,\ \det M_2 = k_1k_2 + (k_2 + \ell_1)\ell_{\text{in}} > 0,$$

and

$$\det M_3 = \alpha k_1 k_2 + \ell_1\ell_2\ell_{\text{in}} > 0\,,\ \det M_4 = \ell_{\text{in}}\mu(\ell_1\ell_2 - \alpha k_2) = r\alpha + s > 0.$$

Thus, the dynamical system moves towards the stationary state from every initial state [25]. In particular, a traumatic change of $\alpha$, which is a decrease in general, leads to a changed stationary state after a transient phase.

The oscillating behavior and the role of the side compartment fat are generally discussed in [3]. We recognize the general behavior in the particular model in the following section dealing with certain scenarios.

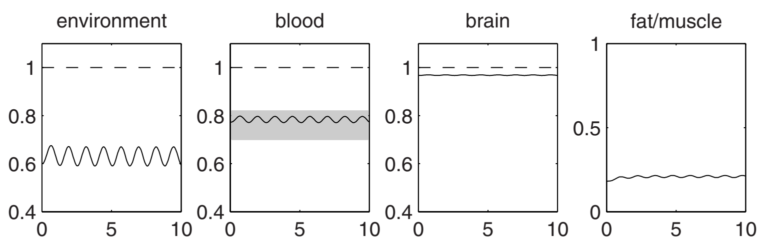



**Fig. 2.** A cyclic food intake generates oscillations in the compartments of the supply chain. The amplitudes are decreasing within the supply chain from the environment into the brain and the set-points are reached in an enforced manner due to the increasing pull component. All set-points are normalized to 1. The physiological range does not necessarily contain the set-point, an example is given in grey for the blood glucose. Remark that the supply chain finishes with the brain, fat is a side compartment.

## 4 Scenarios

The long-term model deals with normalized quantities. All set-points are set to 0 in Eqs. (1–4). That simplifies the mathematical analysis without restriction of the generality because the set-points are changed by an affine transformation. In the present section, the set-points are set to 1 to demonstrate that $E$, $B$, $Y$ and $F$ describe positive energy levels in the respective compartment. Also, the time scales are not identical for the scenarios presented here. However, realistic time scales are the result of a suitable transformation, too.

The first scenario we investigate here is a cyclic food intake, see [14] for modeling social and natural cycling, which lead to a recurrent food offer. Experiments show that the energy level in the brain is constant and independent of exogenous influences [9]. The results shown in Fig. 2 reproduce this experimental observation, whereas the energy level in the near environment and in the blood glucose oscillate in an enforced manner. The fat compartment acts as a store and damps the oscillations. The set-point of the energy levels was normalized to 1, and it is remarked that the physiological range of e. g. the blood glucose does not need to contain the set-point. The regulatory system of the blood glucose is under a permanent load by the energy flows.

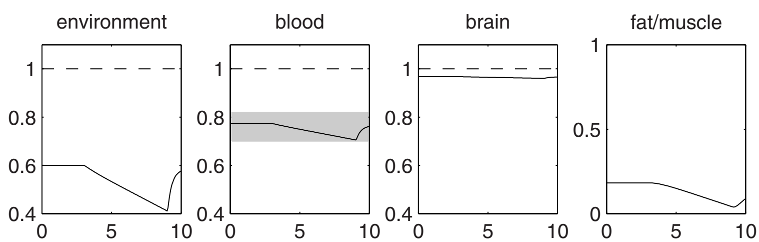



**Fig. 3.** Atrophic situation. The influx into the near environment compartment is cut in a time interval. Consequently, we observe a decrease of the blood glucose and the energy level in the fat compartment. But, the energy level in the brain is nearly unchanged, corresponding to [8].

The next scenario deals with an atrophic situation, i. e. with a lacking food offer. This is modeled by restricting the inflow $j_{\text{in}}$ into the near environment in a time interval. Very old experimental observations show that the energy level in the brain stays nearly constant anyhow [8]. This observation cannot be explained by the glucostatic or lipostatic theory. However, the simulations shown in Fig. 3 indicate the reproduction of the observed behavior. While the energy level in the environment, the blood and the fat decreases dramatically, the energy level in the brain stays nearly constant. The competition for energy is definitely won by the brain [1].

In the last scenario, a traumatic experience occurs, which abruptly decreases the allocation strength $\alpha$. The dynamical system moves towards the new equilibrium with an enlarged fat compartment and an increased blood glucose. Obesity

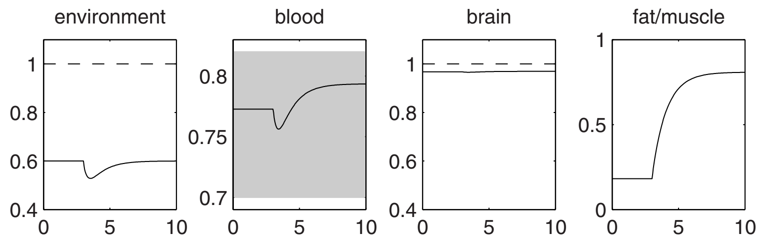



**Fig. 4.** Development of obesity and diabetes. A traumatic experience is modeled by an instantaneous fall of the allocation parameter $\alpha$. More energy is transported into the increasing fat. This enforced transport is a load on the blood glucose, which is decreasing in a shorter time-scale [26]. Finally, the new equilibrium state with an enlarged fat compartment is found. It contains an increase level of blood glucose, i. e., diabetes.

occurs. Within the transient phase, more energy is transported into the fat, and the blood glucose is decreased. That transient decrease has been observed in experiments [26]. Diabetes, i.e. a permanently increased blood glucose, develops at the end of the transient phase when the changed stationary state is nearly reached.

Hence, our model can identify a reason for the development of diabetes, namely the allocation failure. Our explanation shows that a medical decrease of the blood glucose disturbs the energy supply into the brain and leads to the dramatic consequence of a cerebral energy loss [7].

## 5 Conclusion

We have introduced the ideas of the selfish-brain theory, which regards the brain as consumer and as administrative instance in the regulatory system of the human metabolism. In contrast to the glucostatic and lipostatic theory, it concentrates on the pull mechanisms of the brain, which realize the supply of energy on demand. This theory is the first one, which allows us to explain the development of obesity and diabetes.

The investigation of the human metabolism as a regulatory mechanism involves questions of the possibility of a quantitative modeling in a not completely quantified field of research like medicine or life science. The consideration leads to the regard of core models containing a selected number of mechanisms described by simple linear relations. The resulting model properties are independent of the particular choice of the parameters.

We have presented a long-term model of the human metabolism consisting of the four compartments: near environment, blood glucose, brain and periphery, where the periphery combines fat and muscles. The activation of the ventromedial hypothalamus governing the allocation and of the lateral hypothalamus enter the model by the dependency of the fluxes on the brain energy level.

The model allows a rough simulation of different scenarios like cyclic food offer or atrophic periods. While the first scenario leads to oscillating energy levels in the environment, the blood and the fat, but not in the brain, the second scenario results in a nearly constant energy level in the brain, too, but the ones in the other three compartments decrease. These simulation results are in accordance with a number of known experimental observations.

Furthermore, the model simulates the development of obesity and diabetes after a traumatic experience leading to an instantaneous decrease of the allocation strength. It may be followed by a second phase when the changed fluxes damage the self-excited growth of the beta-cells.

The systemic understanding of the human metabolism would enable us to focus on central phenomena and key relations in the model. But until then, the presented model should be discussed in its interaction to other regulatory systems like the stress axis [27] and the memory [13].

## References

1. Peters, A., Schweiger, U., Pellerin, L., Hubold, C., Oltmanns, K.M., Conrad, M., Schultes, B., Born, J., Fehm, H.L.: The selfish brain: competition for energy resources. Neurosci. Biobeh. R 28, 143–180 (2004)
2. Peters, A., Pellerin, L., Dallmann, M.F., Oltmanns, K.M., Schweiger, U., Born, J., Fehm, H.L.: Causes of obesity – looking beyond the hypothalamus. Prog. Neurobiol. 81, 134–143 (2007)
3. Langemann, D.: Selfish brain theory: mathematical challenges in the top-down analysis of metabolic supply chains. In: Grundy, J. (ed.) ER 2007. LNCS, vol. 4801, pp. 39–49. Springer, Heidelberg (2007)
4. Mayer, J.: Glucostatic mechanism of regulation of food intake. N. Engl. J. Med. 249/1, 13–16 (1953)
5. Kennedy, G.C.: The role of depot fat in the hypothalamic control of food intake in the rat. Proc. Royal Soc. Lond. B, Biol. Sci. 140/901, 578–592 (1953)
6. Zhang, Y., Proenca, R., Maffei, M., Barone, M., Leopold, L., Friedman, J.M.: Positional cloning of the mouse obese gene and its human homologue. Nature 372/6505, 425–432 (1994)
7. Couzin, J.: Deaths in diabetes trial challenge a long-held theory. Science 319, 884–885 (2008)
8. Krieger, M.: Über die Atrophie menschlicher Organe bei Inanition [On the human atrophy as a result of undernourishment]. Z. Angew. Anat. Konstitutionsl 7, 87–134 (1921)
9. Oltmanns, K.M., Melchert, U.H., Scholand-Engler, H.G., Howitz, M.C., Schultes, B., Schweiger, U., Hohagen, F., Born, J., Peters, A., Pellerin, L.: Differential energetic response of brain vs.skeletal muscle upon glycemic variations in healthy humans. Am. J. Physiol. Regul. Integr. Comp. Physiol. 294, R 12–R 16 (2008)
10. Magistretti, P.J., Pellerin, L., Rothman, D.L., Shulman, R.G.: Energy on demand. Science 283, 496–497 (1999)
11. Murray, J.D.: Mathematical Biology. Springer, New York (2002)
12. Duarte, N.C., Becker, S.A., Jamshidi, N., Thiele, I., Mo, L.M., Vo, T.D., Srivas, R., Palsson, B.Ø.: Global reconstruction of the human metabolic network based on genomic and bibliomic data. Proc. Natl. Acad. Sci. 104, 1777–1782 (2007)

13. Langemann, D., Pellerin, L., Peters, A.: Making sense of AMPA receptor trafficking by modeling molecular mechanisms of synaptic plasticity. Brain Research 1207, 60–72 (2008)
14. Langemann, D., Peters, A.: Deductive functional assignment of elements in the appetite regulation. J. Biol. Phys. (in print, 2008)
15. Berthoud, H.R.: Mind versus metabolism in the control of food intake and energy balance. Physiol. Behav. 81, 781–793 (2004)
16. Morton, G.J., Cummings, D.E., Baskin, D.G., Barsh, G.S., Schwartz, M.W.: Central nervous system control of food intake and body weight. Nature 443, 289–295 (2006)
17. Neary, N.M., Goldstone, A.P., Bloom, S.R.: Appetite regulation: from the gut to the hypothalamus. Clin. Endocrino. 60, 153–160 (2004)
18. Dallman, M.F., Pecoraro, N., Akana, S.F., la Fleur, S.E., Gomez, F., Houshyar, H., Bell, M.E., Bhatnagar, S., Laugero, K.D., Manalo, S.: Chronic stress and obesity: a new view of comfort food. Proc. Natl. Acad. Sci. 100/20, 11696–11701 (2003)
19. Petrovich, G.D., Setlow, B., Holland, P.C., Gallagher, M.: Amygdalo-hypothalamic circuit allows learned cues to override satiety and promote eating. J. Neurosci. 22, 8748–8753 (2002)
20. Malik, V.S., Schulze, M.B., Hu, F.B.: Intake of sugar-sweetened beverages and weight gain. Am. J. Clin. Nutr. 84, 274–288 (2006)
21. DiMeglio, D.P., Mattes, R.D.: Liquid versus solid carbohydrate: effects on the food intake and body weight. Int. J. Obesity 24/6, 794–800 (2004)
22. Vannucci, S.J., Maher, F., Simpson, I.A.: Glucose transporter proteins in brain: Delivery of glucose neurons and glia. Glia 21, 2–21 (1997)
23. Bouret, S.G., Draper, S.J., Simerly, R.B.: Trophic action of leptin on hypothalamic neurons that regulate feeding. Science 304, 108–110 (2004)
24. Beverly, J.L., de Vries, M.G., Beverly, M.F., Arseneau, L.M.: Norepinephrine mediates glucoprivic-induced increase in GABA in the ventromedial hypothalamus of rats. Am. J. Physiol. Regul. Integr. Comp. Physiol. 279, R 990-R 996 (2000)
25. Hubbard, J.H., West, B.H.: Differential Equations: A Dynamical Systems Approach: Higher-Dimensional Systems. Springer, New York (1995)
26. Bingham, N.C., Anderson, K.K., Reuter, A.L., Stallings, N.R., Parker, K.L.: Selective loss of leptin receptors in the ventromedial hypothalamic nucleus results in increase adiposity and a metabolic syndrome. Endocrinology 149(5), 2138–2148 (2008)
27. Peters, A., Conrad, M., Hubold, C., Schweiger, U., Fischer, B., Fehm, H.L.: The principle of homeostasis in the hypothalamus-pituitary-adrenal system: New insight from positive feedback. Am. J. Physiol. Regul. Integr. Comp. Physiol. 293/1, R 83-R 89 (2007)

# Designing Privacy-Aware Personal Health Record Systems

Reza Samavi and Thodoros Topaloglou

University of Toronto, 5 King's College Road, Toronto, Ontario M5S3G8, Canada
{samavi,thodoros}@mie.utoronto.ca

**Abstract.** Implementation of Personal Health Record (PHR) systems involves multiple stakeholders with different interpretations and expectations; more importantly it involves changes in the custody of data, patient privacy, and consent management. In PHR analysis we need to answer questions such as: Who is the provider of PHR? Who has access to the patient data and why? And how the system can empower the patient? And how can the patient privacy be managed. This paper exploits techniques from Goal and Agent-oriented Requirements Engineering and proposes a methodological framework for dealing with concerns surrounding PHR systems. The framework is illustrated through an example that emphasizes the privacy aspects of PHRs.

## 1 Introduction

Personal Health Record Systems (aka PHR or PHRS) promise to empower patients and make them active participants in their care [1]. A PHR ventures to keep all information related to an individual's health either entered by the individual or interactively pulled from provider's Electronic Health Records (EHR). Therefore a PHR system, and its data, affect not only the patient, but other stakeholders such as payers, providers, and clinicians. These stakeholders are related to each other in a social setting, where at the same time each stakeholder is autonomous with its own goals, motivations and intentions. There are many sensitive issues in the health process with regards to PHRs that need to be investigated. A recent study reports that patients are willing to embrace PHRs if they contribute to their quality and continuity of care, and more importantly if their privacy and security are carefully managed [2]. It is important that PHR designs are well documented and communicated such that "big picture" becomes available and understood by its stakeholders early in the deign process. This necessitates the use of advanced conceptual modelling and early requirement engineering (RE) methodologies.

In PHR analysis we need to answer questions such as: Who is the provider of PHR? Who has access to the patient data and why? How the system can empower the patient while all privacy policies are in the place? And how can we verify that the system to-be for a PHR matches with real needs of its stakeholders? Therefore, thorough analysis of stakeholders' preferences, objectives, intentions, their dependency and trust relationship, and the policies that involved (e.g. HIPAA [3] or PIPEDA [4]) are crucial for the success of PHR systems. Without a systematic analysis of PHR requirements and stakeholders expectations, it is hard to determine how any of the

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 12–21, 2008.

currently proposed technologies meets the needs of PHRs. The use of appropriate requirements modelling languages can create an abstract view of strategic relationship between stakeholders. An agent-oriented requirements modelling framework can bring forth the PHRs stakeholders goals, intentions and motivations while additional modelling mechanisms are needed to describe the coordination of trust relationships, privacy policies, user preferences, and data protection policies. In this work, we apply a combination of i*/Tropos [5] and S(ecure)Tropos [6] agent-oriented modelling frameworks in order to describe the functional dependencies and the trust network of actors in PHRs. We expect that this work will help us to understand the organizational context of a PHR system along with the goals of participants and their social relationships and clarify the functional and privacy requirements of the system to-be.

In Section 2 of this paper we provide a definition and taxonomy of PHRs. In Section 3 we outline our rationale for using agent-oriented modelling language for PHRs requirements analysis. Then we introduce a hypothetical case study and show the power of a requirement engineering framework on PHR analysis. In Section 4, we convert our experience into a practical methodology for PHRs requirements analysis. Finally, we conclude the paper with some directions for future work in Section 5.

## 2 PHR Definition, Characteristics, Types and Taxonomies

PHR is defined as an electronic application through which individuals can access, manage and share their health information in a private, secure and confidential environment [8]. Different models of PHRs have been proposed. From a stand-alone PHR which does not have direct connectivity to the records stored on providers databases to a tethered PHR which is owned and entirely managed by one or more health institution. Patient's control over her data in the latter is limited to viewing a pre-designed portal, while in the former, patient is responsible for entering all the data by herself. Between these two extremes there is the *hybrid PHR* or *interconnected PHR*. Each PHR type is associated with different risks and opportunities both for patient herself and for the other participants in the health care process. For example, in stand-alone PHR, managing privacy is simplified, since the PHR does not have a direct connection to the providers' records. In such a scenario, keeping the PHR up-to-date and patient's ability to correctly and trustworthy enter all medical information on his record are major concerns. By contrast, with interconnected PHRs, the integration of existing data with newly acquired data from provider EHRs, and managing the patient privacy are the major concerns. The PHR could potentially transform the whole health care system including the relationship between patient and the providers. This happens because PHRs empower patients to participate in their own care in ways that were not able to do so far. For example they can use their PHR able them to provide more relevant data to clinicians, or to monitor compliance with a treatment plan, or use the PHR for data exchange between different providers.

## 3 PHR Requirement Analysis

The success or failure of information system development highly depends on the quality of their requirements [9]. The quality of requirements in turn depends on requirement elicitation techniques [10]. An important step in any information system design is selecting an appropriate elicitation technique. For PHR requirements analysis, the modelling technique should be able to capture the multiple goals of PHR stakeholders and detect possible conflicts and inconsistencies among them.

Agent–oriented modelling techniques such as i*/Tropos have been successfully applied in Requirements Engineering problems [11], strategic analysis of business models [12], and modelling trust, security and privacy [13]. The agent abstraction in i* supports the elicitation, exploration, and analysis of the social actors and the systems-in-the-world, possible alternatives in how they relate to each other, and the pros and cons of the alternatives. i* captures characteristics such as intentionality, autonomy, and socialization of all agents in a dependency network. Therefore i* offers the right abstractions to analyze PHR requirements. While a detailed description of i* is beyond the scope of this report, we will describe the parts of the i* notation as they are used in our analysis. We also present a scenario of PHR usage in order to illustrate how i* can be used in the course of PHRs analysis.

### 3.1 A Hypothetical Case Study

Assume that Alice wants to visit a neurologist for the pains she feels in her wrists. While she likes to share with her doctor all her health record, she prefers to avoid sharing attention deficit problem she had experienced during her pre-teenage. However, she doesn't like to entirely delete this part of data from her health record, because she has decided to counsel with a psychiatrist later this month to discuss her poor performance in last semester at the university. Her psychiatrist decides to seek counsel for Alice's case from one of his colleagues in a research institute (RI) specialized in ADHD (attention-deficit hyperactivity disorder). Alice agrees to share her Health data if only the relevant part will be transmitted, her data only be used for her treatment purpose. and the data will be removed from the RI repository as soon as her case is closed.

We use this case study throughout the rest of this paper and show how conceptual modelling can help understand and systematically analyze Alice's goals, her clinicians' goals, and how a PHR system may address these goals.

### 3.2 Modelling PHRs Socio-technical Environment Using i*/Tropos

i*/Tropos is a framework for agent-oriented requirements analysis. Agent modelling aims to determine what various actors want and how (and whether) their objectives are achieved. The name i* stands for distributed intentionality [6], referring to the fact that actors have intentions and that they do not necessarily share common goals. An *actor* (○) interacts with other actors not only through actions or information flows but also relate to each other at an intentional level. They depend (—D—) on each other to achieve *goals* (⬭), perform *Tasks* (⬡), and furnish *Resources* (▭). While each actor aims to pursue his/her strategic goals, they rely on others through a network of intentional dependencies. A *softgoal*(⬭) is a goal without a clear-cut criterion for achievement. Softgoals are typically used to represent quality goals. Additional intentional elements are also used to analyze the reasoning structure within each actor. *Means-ends* (⟶) shows a particular way to achieve a goal. *Decomposition* (+—)

shows how an intentional element is decomposed. *Contribution* ( ) shows a contribution toward satisfying a soft goal. A *role* ( ) conveys the notion of an abstract actor. An *agent* ( ) is a concrete, physical actor. A role can be played by one or more agents. Using the i* notation one can describe a system and its socio-technical environment.

The first step in using i* for requirements analysis is Actor modelling. In the case outlined above, Alice, the neurologist, and the psychiatrist are the agents. In an abstract level they can be represented by *actor* such as *Patient* and *Clinician*. Usually actors have specific goals that they like to fulfill. If they are not capable to fulfill them individually, they depend on other actors in the system to fulfill these goals. Figure 1 shows a much simplified Strategic Dependency (SD) model between actors in our example. The domain requirement analysis with i* is an iterative and semi-automatic process. Modelling starts with a basic idea about the domain. As we add more constraints and dependencies to the model, we begin to explore the vulnerabilities of depender since in each dependency relationship the dependee may fail to fulfill a goal or a task. Therefore, the requirement for alternative ways of performing a task or achieving a goal might be identified.  Explicit representation of goals and dependencies may unravel new actors and goals (Patient Medical Record or PMR in Figure 1). A Strategic Dependency (SD) model expresses what actors want from each others in a high abstract level. We use this model in order to explore how the actor's expectations may change, for instance, by adopting new technologies, or using a new software agent in the system. In this way we analyze the requirements for the new system (to-be).

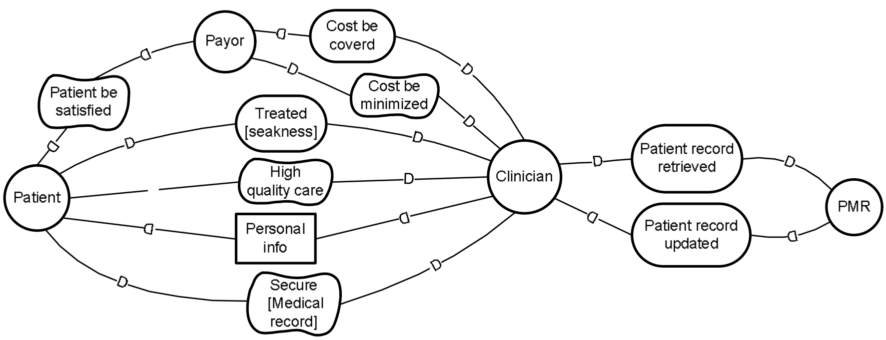



**Fig. 1.** A simplified Strategic Dependency model when provider uses a PMR system

In the case study Alice visits two *Clinicians*, a *Neurologist (C1)* and a *Psychiatrist (C2)*. The model of Figure 2 represents this with two actor instantiation relationships of the *Clinician* role. In the new landscape, the *Psychiatrist* (C2) not only depends on *patient* for her *personal info*, but he depends on *Neurologist* (C1) for *up-to-date personal health data* of Alice. In such a scenario, if C1 does not provide the resource on time, it makes the patient vulnerable and put her heath at risk. This simple example clearly shows that while Alice is the owner of her health data, the system can make her vulnerable. The i* SD model makes this implicit vulnerability explicit.

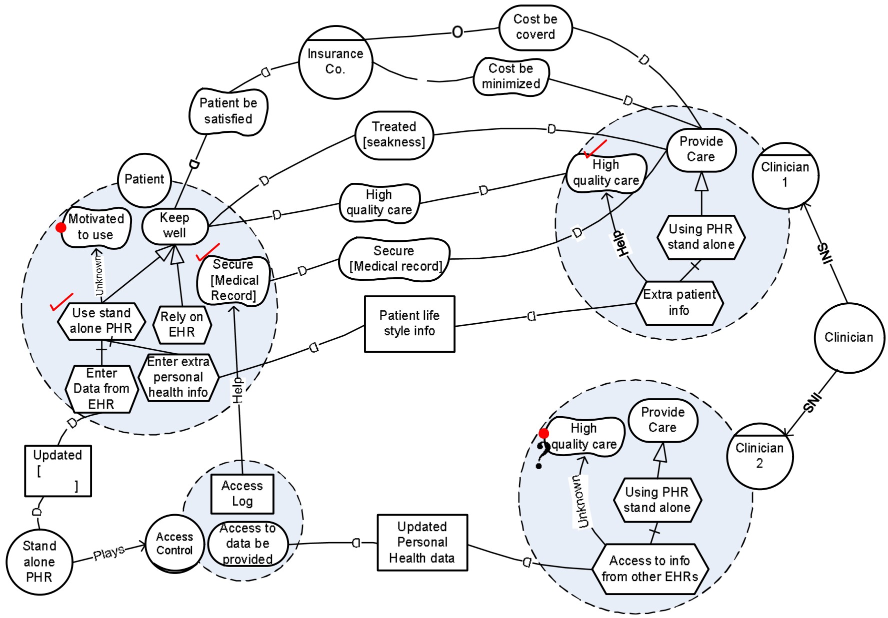



**Fig. 2.** i*Tropos Rationale Model when patient uses a stand alone PHR system

To eliminate this vulnerability another alternative could be to use an interoperable EHR controlled by a third party (e.g, the healthcare system[1]). Because of space limitations, we do not present the model for this alternative, but our analysis revealed that this creates another chain of dependencies which could be a source of additional efforts and inefficiencies. A third alternative is that the patient uses a stand alone PHR. In this option, Alice stores all her health data on a PHR and she personally keeps it up-to-date. So, when she meets *Psychiatrist* she has already entered previous data from her visit with *Neurologist.* It seems that a stand-alone PHR can potentially play the role of an interoperable EHR. Strategic Rationale (SR) model in i*/Tropos helps investigate this scenario (Figure 2).

SR model in i*/Tropos provides semantics for the internal goal structure for each actor. This model helps analysts to construct and explore the space of alternatives available to each actor. Graph-based algorithms, such as qualitative label propagation, can be applied to interactively evaluate whether goals are achieved. Five type of qualitative i* labels *satisficed* ✓, *denied* ✗, *weakly satisficed* ✓., *weakly denied* ✗., *unknown* ?, and *conflict* ⚡ are used for this purpose. For example, in this scenario (replacing EHR with standalone PHR) we notice that two softgoals, *Secure Medical Record* for *Patient* and *High quality of care* for C1 are satisficed, while the status for two other softgoals, *Motivated to use* of the Patient and *High quality care* of C2 are unknown. These two goals need further refinement. We apply the layering technique proposed in [12] and NFR framework [7] for the refinement of these strategic goals (Figure 3).

[1] Canada Health Infoway is such an example.

### 3.3 Modelling PHRs Stakeholders Strategic Goals

The purpose of using goal models is to understand the degree at which a design choice is in synergy (or conflict) with top strategic goals of an actor. The NFR framework used for goal modelling was originally developed for dealing with non-functional requirements in software engineering [7]. The NFR consists of defining a set of softgoals which express criteria for system quality, and evaluating the degree to which an alternative model structure would lead to an information system (IS) that satisfices those qualitative goals. Links between top level qualitative goals to alternative models for IS creates a softgoal interdependency graph (SIG). A goal graph provides vertical traceability from high-level strategic concerns to low-level technical details [14].

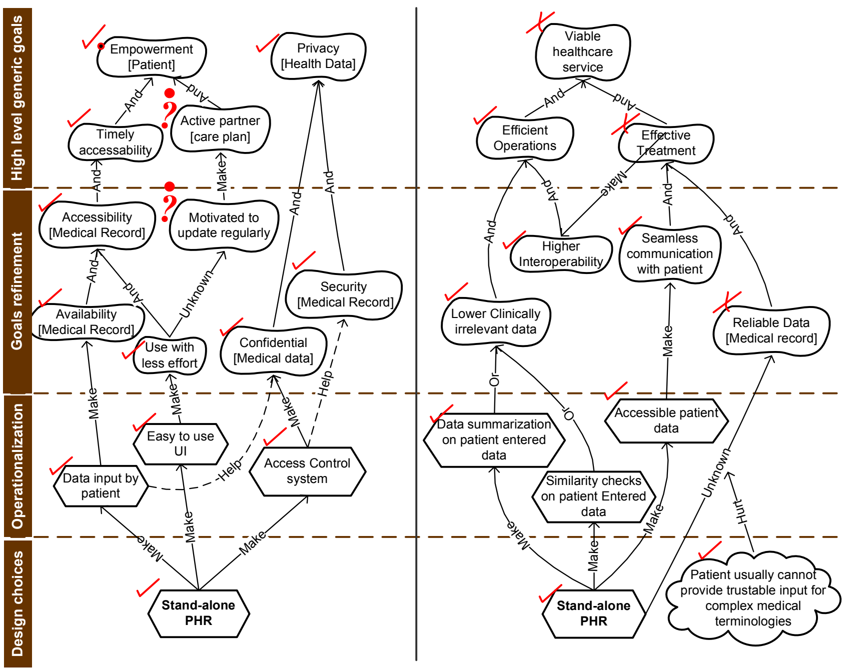



**Fig. 3.**(a) Patient's goal model (b)Clinician's goal model

In our running example, we first build a simplified SIG for the Patient (Alice). The high level strategic goals for the *Patient* are identified as *Patient empowerment* and *Patient privacy*. The design choice at the bottom of the graph is *Standalone PHR* (Figure 3-a). As moving downward (top-down) in SIG, the generic elements on top can be ***refined*** to some elements which are contextual and domain dependent. At the end they can be ***operationalized*** to processes and mechanisms which are implementable. We use the decomposition relationship for this purpose. Conversely, when moving upward (bottom-up), the graph shows the contribution and correlation of design choices towards the refined goals on top. In this way, the goal model makes the requirements for the stand-alone PHR system explicit. For example, we identified three requirements mentioned in the operationalization section of Figure 3-a; (i) Data input by patient, (ii) Easy to use GUI, and (iii) Access control System from refinement and operationalization of high level goals.

Patient's decision on her design choices may affect other stakeholders' goals. Therefore, goal modelling can not be conducted for each actor independently. As such, the decisions made by other actors or other important assumptions and beliefs will be introduced in model as argumentation construct (little cloud in Figure 3-b). Figure 3-b represents the Clinicians goal model. The goal evaluation in this case shows *Reliable Data for Medical Record* may not be satificed due to the asserted assumption that a *Patient usually cannot provide trustable input for complex medical terminologies.* The expressiveness of the model makes such assumptions explicit.

The whole process of requirement analysis is iterative; we may go back and forth a few times between different stakeholders' goal model and SD and SR models in order to verify and validate the elicited requirements. As we continue the agent-oriented modelling exercise we find answer to our questions and identify new requirements.

### 3.4 Requirements Analysis for Privacy-Aware PHRs

PRIME [15] defines information privacy as "being in a position to determine for oneself, when, how, and to what extent information about oneself is communicated to others." In the case of PHRs, a patient as the owner of his/her health data should be in a position to enforce his/her privacy in any instance of sharing information with other parties involved in the healthcare process. All privacy policies (e.g. HIPAA [3], PIMEDA [4]) acknowledge the patient's rights to delegate, monitor or specify purpose when his/her data are shared with others. These policies need to be operationalized by the specific requirements in the system-to-be.

Until now, the proposed modelling framework helps to analyze functional requirements of PHRs alongside the privacy as a high level strategic goal of the patient. This allowed us to explicitly identify how a design choice for PHR may affect this goal. In the model shown, in Figure 3-a access control considered as a requirement for achieving the privacy goal. Here we observe a leap between the high level privacy goal and its next refinement as Access control [17]. While Access control deals only with permission and denial of access, privacy protection requires dealing with ownership of data, purpose of access, obligations of data recipients, and monitoring the compliance with the promised policies [16]. Therefore, we choose to utilize STropos [6] which is an extension of the i*/Tropos for privacy requirements analysis.

The extended methodology provides two new modelling opportunities, *Trust modelling* and *Delegation modelling*. The former differentiates between actors which *trust* other actors for goal, tasks, and resources, with actors which *own* them. The later differentiates between actors that delegate to other actors *permission* (you can do but you do not need to [18]) and actors that delegate *execution* on goals, tasks, and resources (if you do not do the job, I hurt). This enhanced modelling technique incorporates concepts such as *ownership* of goal, task, or a resource; *trust* between two actors when *truster* trust to *trustee* on the *trusteum*; *provisioning* which indicates the capability of an actor to achieve some goal, or execute a task; and *delegation* between two actors when *delegater* delegates to *delegatee* permission or execution on the *delegatum.*

Figure 4-a depicts the trust relationship between Alice, her Psychiatrist and the Research Institute. The Trust model captures the owners of goals, tasks, or resources, and drives the incorporation of privacy requirements into the functional requirements of the PHRs [6]. Alice owns the resource PHR, and the doctor owns the treatment goal. The owner of a goal or resource has full authority of the fulfillment of a goal or use of a resource. The edge between goal and the actor is labeled by **O**. Alice trusts her doctor, therefore the edge between these two actors labeled by **T**. The trust between Psychiatrist and the Research Institute is also depicted in Figure 4-a. Interestingly, these two actors trust each other on a resource that neither of them owns. The model so far is descriptive in terms of unraveling the trust between actors. For example, the model explicitly shows that there is not a trust relationship between Alice and Research Institute. The model can be prescriptive in terms of identifying privacy requirements. In the model shown in Figure 4- Alice delegates usage on her PHR to her doctor. This delegation is a *delegation for execution* of a goal and labeled by **De**. The doctor in order to fulfill his goal (Alice's Treatment) wants to delegate *Patient info* to the *Research Institute*. Since Doctor is not the owner of the resource, a functional requirement must be introduced: Doctor needs Alice's consent for delegation. The type of delegation here is *delegation of permission* and labeled by **Dp**. If RI uses the resource, it should be for the purpose mentioned in the permission only.

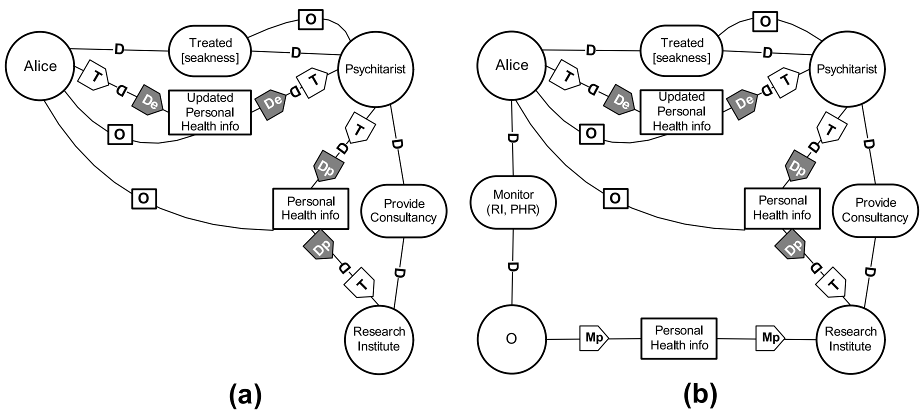



**Fig. 4.** (a) Partial trust and functional dependency model capturing delegation. (b) Partial trust model and functional dependency model capturing monitoring.

Figure 4-b depicts the same situation, but this time Alice distrusts the Research Institute and would like to enforce her privacy. She hires an ombudsman (O) to monitor the Research Institute records to ensure the usage purpose and the deletion of record after the specified retention period. The functional dependency between Alice and the ombudsman is captured by the *Monitor (Research Institute, PHR)* goal. The edge between Ombudsman and Research Institute represents the monitoring on permission which is labeled by **Mp**.

# 4 A Methodology for the PHR Requirements Analysis

We now discuss our proposed methodology for requirements analysis of Personal Health Record systems. In our proposal, the i*/Tropos, and Secure Tropos are two complementary agent oriented frameworks used to model the functional and privacy concerns of PHRs. The NFR framework is also used for building and evaluating each actor's strategic goals. Figure 5 outlines a set of practical steps, using these three frameworks, to systematically analyze the PHR requirements. As illustrated during this paper, we start with the modelling actors in i*/Tropos, then modelling of their dependencies and the creation of a Strategic Rationale model. For each actor in the SR model we build the goal graph to ensure the alternatives selected in the rationale model are in line with the strategic goals of each actor. This is an iterative process and we have to go back and forth between goal models of different actors, change the assumptions and evaluate the goal model again. The next step will be modelling delegation and trust. A conceptual refinement of the delegation and trust comes later; this allows to capture and model important Privacy concerns [18].

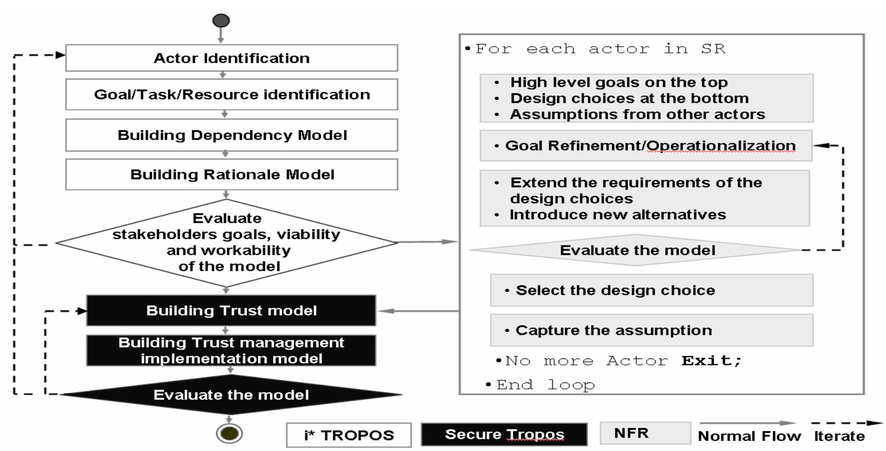



**Fig. 5.** The proposed methodology for PHRs requirements analysis

# 5 Conclusions

Emerging technologies of PHRs surface concerns about their connectivity to EHRs and patient privacy and security. We have showed that PHRs are complex and socio-technical systems that involve different stakeholders. We have provided evidence that because of this complexity a systematic requirements elicitation methodology in early stage of PHRs design is required. None of the requirements engineering techniques individually can sufficiently address requirements analysis of an ideal PHR system. We proposed a methodology that is as hybrid of i*/Tropos, STropos and NFR frameworks for this purpose. The proposed methodology helped to understand important properties of current PHRs technologies. We applied the above mentioned agent-oriented frameworks to create conceptual models that describe a PHR system ("as-is" state). Using a hypothetical case study we addressed different privacy concerns which translate to new requirements for PHRs ("to be" state). The next step of our work is to extend our modelling efforts and elicit a complete list of privacy requirements for PHRs based on reality derived scenaria concerning privacy.

## References


1. Tang, P.C., Lansky, D.: The missing link: bridging the patient-provider health information gap. Health Aff. 24, 1290–1295 (2005)
2. Halamka, J.D., Mandl, K.D., Tang, P.C.: Early Experiences with Personal Health Records. J. Am. Med. Inform. Assoc. 15(1), 1–7 (2008)
3. HIPAA., Act of 1996 (accessed, May 2008), http://aspe.hhs.gov/admnsimp/pl104191.htm
4. PIPEDA., (accessed, May 2008), http://www.privcom.gc.ca/legislation/02_06_01_e.asp
5. Bresciani, P., Giorgini, P., Giunchiglia, F., Mylopoulos, J., Perini, A.: Tropos: An Agent-Oriented Software Development Methodology, vol. 8(3), pp. 203–236 (2004)
6. Giorgini, P., Massacci, F., Mylopoulous, J., Zannone, N.: Requirements Engineering meets Trust Management. In: Jensen, C., Poslad, S., Dimitrakos, T. (eds.) iTrust 2004. LNCS, vol. 2995, pp. 176–190. Springer, Heidelberg (2004)
7. Chung, L., Nixon, B., Yu, E., Mylopoulos, J.: Non-Functional Requirements in Software Engineering. Kluwer Academic Publishers, Dordrecht (2000)
8. Markle Foundation: Connecting for Health. Connecting Americans to their healthcare. Final rep. of the wg on policies for electronic information sharing. NY Markle Found (2004)
9. Jones, C.: Patterns of software failure and Success. Thomson (1996)
10. Hickey, A., Davis, A.: The role of Requirement Elicitation technique in Achieving Software Quality. In: Req. Eng. WS: Foundation for Software Quality (REFSQ) (2002)
11. Yu, E.: Towards Modelling and reasoning Support for Early-Phase Requirements Engineering. In: Proc. RE 1997-3rd Int. Symp. on RE, Annapolis, pp. 226–235 (1997)
12. Samavi, R., Yu, E., Topaloglou, T.: Strategic Reasoning about Business Models: a Conceptual Modelling approach. J. Info. Sys. & E-Business Manag. (2008)
13. Liu, L., Yu, E.: Designing Information Systems in Social Context: A Goal and Scenario Modelling Approach. J. Information Systems 29(2), 187–203 (2002)
14. van Lamsweerde, A.: Goal-Oriented Requirements Engineering: A Guided Tour. Invited minitutorial. In: Proc. RE 2001, Int. Joint Conf. on RE, Toronto, pp. 249–263. IEEE, Los Alamitos (2001)
15. PRIME: Privacy and Identity Management for Europe (accessed, May 2008), https://www.prime-project.eu/
16. Karjoth, G., Schunter, M.: A Privacy Policy Model for Enterprises. In: Proc. Of CSFW 2002, pp. 271–281. IEEE Press, Los Alamitos (2002)
17. Giorgini, P., Massacci, F., Zannone, N.: Security and Trust Requirements Engineering. In: Aldini, A., Gorrieri, R., Martinelli, F. (eds.) FOSAD 2005. LNCS, vol. 3655, pp. 237–272. Springer, Heidelberg (2005)
18. Giorgini, P., Massacci, F., Mylopoulous, J., Zannone, N.: Modelling Security Requirements Through Ownership, Permission and Delegation. In: Proc. of the 13th IEEE Requirements Engineering Conference (RE 2005) (2005)

# Linking Biological Databases Semantically for Knowledge Discovery*

Sudha Ram, Kunpeng Zhang, and Wei Wei

Department of MIS
Eller College of Management
University of Arizona
Tucson, AZ 85721
{sram,kpzhang,wwei}@email.arizona.edu

**Abstract.** Many important life sciences questions are aimed at studying the relationships and interactions between biological functions/processes and biological entities such as genes. The answers may be found by examining diverse types of biological/genomic databases. Finding these answers, however, requires accessing, and retrieving data, from diverse biological data sources. More importantly, sophisticated knowledge discovery processes involve traversing through large numbers of inherent links among various data sources. Currently, the links among data are either implemented as hyperlinks without explicitly indicating their meanings and labels, or hidden in a seemingly simple text format. Consequently, biologists spend numerous hours identifying potentially useful links and following each lead manually, which is time-consuming and error-prone. Our research is aimed at constructing semantic relationships among all biological entities. We have designed a semantic model to categorize and formally define the links. By incorporating ontologies such as Gene or Sequence ontology, we propose techniques to analyze the links embedded within and among data records, to explicitly label their semantics, and to facilitate link traversal, querying, and data sharing. Users may then ask complicated and ad hoc questions and even design their own workflow to support their knowledge discovery processes. In addition, we have performed an empirical analysis to demonstrate that our method can not only improve the efficiency of querying multiple databases, but also yield more useful information.

**Keywords:** Semantics, Conceptual modeling, Ontology.

## 1 Introduction

Understanding the relationships and interactions between biological functions/ processes and biological entities has become a very important research focus in the life sciences. Diverse biological data are being collected and stored to enable retrieval, sharing, and scientific discovery. Life science researchers use these data to discover implicit or explicit knowledge by traversing through a large numbers of inherent links

* This research is supported in part by research grants #EF0735191 and #IIS0455993 from the National Science Foundation, USA.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 22–32, 2008.

among the various data sources. Currently, the links among data are either implemented as hyperlinks without explicitly indicating their meanings and labels, or hidden in a seemingly simple text format. Consequently, biologists spend numerous hours identifying potentially useful links and following each lead manually, which is time consuming and error-prone.

The three major repositories NCBI (National Center for Biotechnology Information), DDBJ (DNA Databank of Japan), and EBI (European Bioinformatics Institute) have made significant efforts recently to provide integrated access to multiple biological databases, e.g. via the Entrez system[4,5]. Such systems are useful because they provide access to multiple databases through a common user interface. However, they typically do not attempt to label the links by understanding and representing their semantics. Thus users are still forced to traverse each link individually to see if they are worth exploring. Some work has been done to understand semantic links between biological databases [13,15,16,17] and to label them. However this work is limited to a subset of data including publications, diseases and genes. To the best of our knowledge, past work on conceptual models for biology do not define new constructs to explicitly represent their semantics.

In this paper, we describe a semantic model for biology data, including biological functions, processes, and sequences. We also show how the relationships among these concepts can be used to explicitly label the hidden links among different databases. In addition, we demonstrate how these link semantics can be extracted using text recognition methods and how they enable knowledge discovery. The rest of the paper is structured as follows. In the first section we review related research in detail and discuss their pros and cons. In the next two sections we define and illustrate our semantic model, and develop a link categorization framework based on this model. After this we present a prototype system that implements the model and semantic links and summarize the results of an empirical evaluation of our approach. Finally, we conclude with a discussion of the directions for future work.

## 2 Related Work

Various approaches to semantically link biological data sources have been proposed by the bioinformatics community. These can be characterized into three major types: (a) some are aimed at understanding how to extract and label links to capture semantic knowledge; (b) others understand the relationship among biological entities in specific domains such as protein interaction, gene regulation, and gene deletions; and (c) some researchers focus on modeling relationships of biological data entities, including biological functions, processes, and sequences.

The Entrez system [4] from the National Center for Biotechnology Information (NCBI) is an integrated, text-based search and retrieval system for several major life sciences databases, including PubMed, Nucleotides and Protein Sequences, Protein Structures, Complete Genomes, Taxonomy, and others. The Entrez system essentially allows navigation from an entry in one database to another in a different database via hyperlinks. However when these hyperlinks are analyzed, the source and the target of a link between biological data entries are potentially at a finer level of granularity, and may correspond to specific sub-elements or fields. Heymann et.al.,[13] consequently develop a methodology to enhance the structure and the meanings associated with these links. They use syntactic and semantic knowledge to explore the explicit meanings of data sources and their interconnections. Similar work on extracting, generating, and labeling links to enhance their semantics is also reported in Lash et. al.,[16] who developed a machine assisted tool to extract and label markers for PubMed abstracts and their links to the human genome. Lacroix et. al.,[15] have further extended this work to propose a framework consisting of an ontological layer, a physical layer, and a data layer to link multiple biological databases. Recently, Lee et. al.,[17] have developed a software tool called LSLink to identify meaningful links, but this tool deals with a subset of databases including OMIM (Online Mendelian Inheritance in Man) (for diseases), PubMed (publications), and GenBank(for genes).

Other researchers such as Paton, et. al.,[9,11,18,23] use a UML model to describe eukaryotic genome sequence data and genome organization, plus a number of important functional datasets, namely protein interaction data, transcription data, and results from gene deletions. Additionally, Helden et. al.,[14] propose a new data model to describe the physical and functional interactions between genes and proteins, including some biological processes. However, they do not explicitly represent their semantics but only label their relationships.

In addition, some related research in the area of understanding the semantics of biology data has also been ongoing for a few years. These include Ram and Wei [19], which develops a semantic model of 3D protein structures, and [20, 21], which develops a model to represent the semantics of DNA sequences and primary protein sequences. In our paper, we build on this semantics-related work to develop a comprehensive conceptual model to represent the semantics of biological entities and their relationships, and show how to use this model to classify and label links among the databases in the Entrez system.

# 3 Our Approach

## 3.1 BioSem: A Semantic Model for Biology

Figure 1 shows a schema using our proposed semantic model called BioSem to represent biological entities, and their relationships. BioSem defines new constructs to express biological data semantics, including compositional relationships among entities, and important biological functions and processes. These constructs are necessary to define a framework for expressing the semantics of links between biological data. This model has been developed based on an understanding of the relationships between biological entities. We provide examples to explain each construct proposed in our semantic model. The schema in Figure 1 represents the semantics of all the data accessible via the Entrez system.

The BioSem model is an extension of the entity relationship model first proposed by Chen[10]. It is different from traditional ER model in that it creates new constructs to describe the semantics of biological entities and their relationships. In biology, sequence is a very important concept. Sequences are of various types, including DNA, genes, proteins, and mRNA. Each of these sequences is composed of atomic components such as nucleic or amino acids. Sequences are related to each other through different types of relationships. One major way to relate them is using compositional relationships which are of two types – sequential aggregation and segment. For example, a) a biological sequence consists of individual *atoms*; b) a Gene is a part of a DNA sequence; c) a protein contains multiple segments called *domains*.

Other types of relationships represent important biological processes and functions, for example, a) a DNA sequence can be *transcribed* and *translated* into a Protein; b) Genes *participate* in regulatory processes; c) Proteins can *interact* with other Proteins; and d) DNA sequences *bind* with specific Proteins. These constructs with specific notations have different semantics, and cannot be represented with the traditional or extended ER models. An early form of Sequential Aggregation has been proposed in the work by Ram and Wei [16]. Segments, Component Binding, Domain Interaction, and Regulatory Aggregation are all new constructs proposed and defined in this paper. Biological publications and diseases are also very important entity classes represented using standard ER modelling constructs.

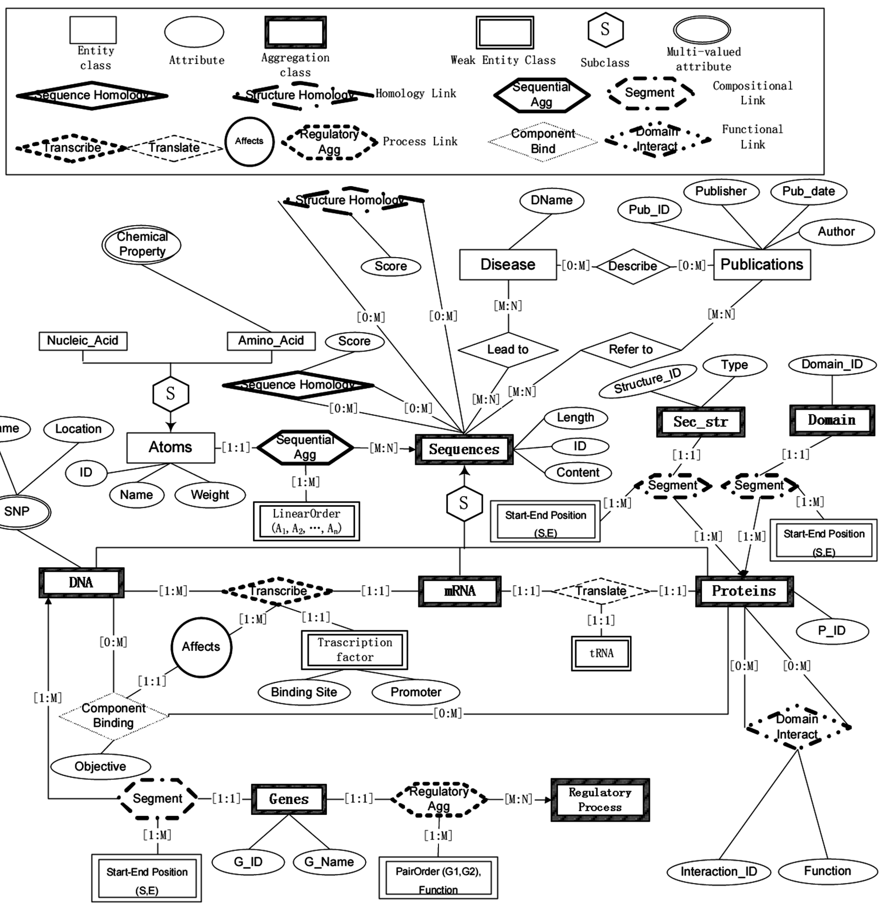



**Fig. 1.** BioSem-Semantic Model of Biological Entities and Their Relationships

**Sequential Aggregation.** The sequential aggregation relationship is defined to represent the relationship between atoms and sequences. A protein sequence is composed of up to 20 types of amino acids. There are 4 types of nucleic acids that are used to compose DNA and mRNA sequences. Each atom appears in a specific position in a sequence and may be repeated. We use $S_{seq}$ to denote this relationship, where $S_{seq}$ is a set of 2-tuple <atom, position>. $S_{seq} = \{<a, p_i> \mid 1 \leq i \leq m\}$, where a is an atom, $p_i$ is its position; m is the length of the sequence. For example, a DNA sequence (AAACGG…TCTT) which consists of 100 bases, is represented as {<A, 1>, <A, 2>, <A, 3>, <C, 4>, <C, 98>, <G, 5>, <G, 6>, …, <T, 97>, <T, 99>, <T, 100>}. The other sequences are similar to DNA, except they may consist of other types of atoms.

**Segment.** This is a kind of part-whole relationship but in the reverse direction. It differs from a sequential aggregation in that it only requires two positions (a start and an end) to specify an entity which is a segment of another entity. For example, a protein may contain segments called domains; a gene is a segment of a DNA sequence. We employ $S_{seg}$ to represent this relationship, where $S_{seg}$ is a set of 3-tuple <Start Point, End Point, Segment Name>. $S_{seg} = \{<s_i, e_i, n_i> \mid 1 \leq i \leq m\}$, where $s_i$ and $e_i$ are start and end positions of the *i*th segment; $n_i$ is the name of the *i*th segment; m is the number of segments. Consider proteins and domains for example. Assume that one protein A has 3 different domains $d_1$, $d_2$, $d_3$ whose start points are $s_1$, $s_2$, $s_3$, end points are $e_1$, $e_2$, $e_3$, and domain names are $n_1$, $n_2$, $n_3$ respectively. We can use start-end positions and domain names to represent this, $\{<s_1, e_1, n_1>, <s_2, e_2, n_2>, <s_3, e_3, n_3>\}$. The relationship between Genes and DNA sequences is similar.

**Component Binding.** A DNA sequence can *transcribe* and *translate* into a protein sequence. During this process, other proteins can bind with the DNA sequence. This relationship is called "component binding". It is important to identify the exact site in the protein where this binding occurs and the nucleic acids in the DNA sequence which participate in this binding. We define $S_{bind}$ to show this binding relationship, where $S_{bind}$ is a set of 5-tuple <protein, DNA sequence, {positions of binding sites in the protein}, {positions of binding sites in the DNA sequence}, objective>. $S_{bind} = \{<p, d, \{a_i \mid 1 \leq i \leq m\}, \{b_j \mid 1 \leq j \leq n,\}, o>\}$, where p is the protein id, d is the DNA sequence id, m is the total number of amino acids taking part in the binding, n is the total number of bases taking part in the binding, $a_i$ is the position of the *i*th binding site in the protein, $b_j$ is the position of the *j*th binding site in the DNA sequence, o is the objective of this binding. Here, binding sites on a protein can be either continuous or discrete. The component binding relationship "*Affects*" the transcription process as shown in figure 1.

**Domain Interaction.** Proteins can interact with each other. For example, some diseases are caused by a protein's malfunction which may be caused by interaction with some other protein. We use $S_{interact}$ to denote the domain interaction relationship, where $S_{interact}$ is a set of 2-tuple, <{protein, protein, domain, domain}, function>. $S_{interact} = \{< \{pa, pb, d_i, d_j\}, f > \mid 1 \leq i \leq m, 1 \leq j \leq n\}$, where pa, pb are the protein ids, $d_i$ is a domain of protein pa, $d_j$ is a domain of protein pb, m is the number of domains of protein pa, n is the number of domains from protein pb. For example, CD4(a kind of antibody) interacts with the shell protein of HIV with a function of activating immune response.

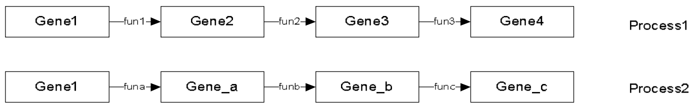



**Fig. 2.** Gene Regulatory Process

**Regulatory Aggregation.** Gene expressions and gene regulations are very important functions because they pass genetic information on from DNA sequences to proteins. The process is actually carried out by a protein, which is a product of a gene translation. Figure 2 shows an example of a regulatory process which is a sequence of ordered gene pairs. In each gene pair, the first gene regulates the function of the second gene.

Regulatory process $S_{reg}$ is a collection of ordered gene pairs. We employ $S_{reg}$ to represent this relationship, where $S_{reg}$ is a set of sets of 2-tuple consisting of < {(gene, gene), function} >. $S_{reg} = \{\{(g_i, g_j), f_l\} \mid 1 \leq i, i<j, \text{ and } j \leq m\}$, m is the total number of genes, l is the function number. For example, the process1 can be represented as $S_{reg} = \{((g_1, g_2), f_1), ((g_2, g_3), f_2), ((g_3, g_4), f_3)\}$.

### 3.2 Labeling the Links: Link Ontology

The semantic constructs defined in BioSem can be used to characterize and classify the links among biological entities. The hyperlinks recorded in the Entrez system between entries in different databases have the relationship semantics represented in BioSem. However, many of these are hidden in the records or summaries and need to be generated from the hyperlinks and explicitly labeled. For example, in the Entrez system, DNA sequences and genes have hyperlinks to each other. Other links are not even available as hyperlinks, but need to be generated and labeled. For example, the protein-protein interaction link does not explicitly exist in the Entrez system but can be derived and generated using other links in BioSem. In the next sections, we show how these links can be analyzed, and explicitly labeled and stored in a structured format and reused. We classify and label the links using our link ontology which is derived from the relationships in BioSem. These labeled links with semantics can then help users answer complex queries.

The relationships defined in BioSem can be classified into six primary types. These are: compositional, functional, process, homology, document, and cause-effect. These are defined further in this section.

**Compositional Link.** This refers to a "part-whole" relationship such as entity A contains entities B and C. It may indicate where exactly B and C are located in A, or, the relative positions of B and C in A. In BioSem, there are two specific compositional relationships: "Sequential aggregation" and "Segment". Segment indicates where one entity starts and where it ends in the other entity. For example, in the segment relationship between DNA and Genes, one DNA could have many different genes in different segments. Similarly protein secondary structures and protein domains have a "segment link". Sequential aggregation refers to the position of one entity within an aggregate entity. For example, a DNA sequence may consist of many nucleic acids.

**Functional Link.** There are numerous functions performed by a biological system. For example, protein binding with DNA is very common, especially in the process of transcription. This link is used to track which sites (binding sites) in a protein bind with a DNA. Similarly in protein-protein interactions, it is important to track which domain in one protein interacts with which domain in another protein. There are two kinds of Functional links: domain interaction and component binding.

**Process Link.** Biological processes are very complex and a source of life. In our work, we mainly focus on three types of processes: transcription, translation, and gene regulation. Transcription and translation processes link DNA sequences and Proteins, while regulation processes link pairs of genes.

**Homology Link.** Computational algorithms (BLAST, pBLAST, etc.) are widely used to perform sequence or structural homology searches. Similarities among sequences and structures can be established by computational methods if the results suggest significant similarities among them, even when no experimental work has indicated they are related. This feature is very important in the knowledge discovery process since the essence of the process is to detect potential links and to try to explain or reason about them instead of studying things that have already been found. In our model, we consider both sequence and structural homology relationships and label them as homology links.

**Document Link.** Genomics research has resulted in a large number of publications. Our model has the ability to help users locate publications related to a specific biological entity. The document link derived from the relationship between PubMed and sequences in our model is also represented as a hyperlink in the Entrez system.

**Cause-Effect Link.** The cause-effect relationship between biological sequences and diseases is very important for medical diagnosis. This link tracks which disease is related to which gene or protein.

A summary of our classification is shown in the link ontology in Figure 3.

1. **Compositional Link**
   - ◆ *AtomSequence*
     - ● Nucleic-Acid←→ DNA
     - ● Amino-Acid←→ Protein
   - ◆ *Gene←→ DNA*
   - ◆ *Domain←→ Protein*
   - ◆ *Secondary Structure←→ Protein*
2. **Functional Link**
   - ◆ *Domain Interaction*
     - ● Protein←→ Protein
   - ◆ *Component Binding*
     - ● DNA←→ Protein
3. **Process Link**
   - ◆ *Transcription/Translation*
     - ● DNA←→ mRNA
     - ● mRNA←→ Protein
   - ◆ *Gene Regulation*
     - ● Gene←→ Gene
4. **Homology Link**
   - ◆ *Sequence←→Sequence*
     - ● DNA←→DNA
     - ● Gene←→Gene
     - ● Protein←→Protein
   - ◆ *Structure←→Structure*
     - ● DNA←→DNA
     - ● Gene←→Gene
     - ● Protein←→Protein
5. **Document Link**
   - ◆ *Disease←→Publication*
   - ◆ *Sequence←→Publication*
     - ● DNA←→PubMed
     - ● Gene←→PubMed
     - ● Protein←→PubMed
6. **Cause-Effect Link**
   - ◆ *Sequence←→Disease*
     - ● DNA←→Disease
     - ● Gene←→Disease
     - ● Protein←→Disease

**Fig. 3.** Link Ontology

## 4 BioLink: A Prototype System for Linking Biological Databases

We have implemented a prototype system called BioLink[22] using BioSem and the link ontology (see Figure 4).

In this prototype we focus on four primary types of data sources: Nucleotides, Proteins, Genes, and PubMed. BioLink uses four main application programming interfaces available through Entrez: these are Esearch, Esummary, Efetch, and Elink[5]. When a user inputs one or more search criteria using a web based GUI, BioLink calls on, ESearch which uses query keywords to retrieve Unique Identifiers (UIDs) for records satisfying the search criteria. The resulting UIDs are passed on to ESummary, which generates short descriptions of data records with key data fields. EFetch then retrieves the set of text files including publications for given UIDs. ELink returns the set of identifiers for other records that are potentially related to the UIDs. The Annotation Analyzer is at the core of our system, which is responsible for analyzing the hyperlinks returned via Entrez. It uses our link ontology, and other ontologies such as Gene ontology[7], Sequence ontology[12] and UMLS[6] to generate labels for all the hyperlinks and their components. It also generates new links and their labels. All generated links are named and stored in the LINK database. Users may get answers to their queries either from preexisting links stored in the LINK database, or they may be generated dynamically. In addition, we also use APIs provided by BIND[8] which provides some information about protein-protein interaction.

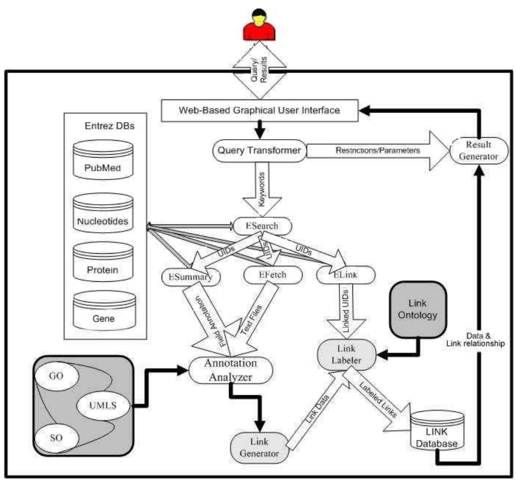



**Fig. 4.** Architecture of BioLink

## 5 Evaluation and Queries

The main goal of BioLink is to make ad-hoc queries more efficient and help in navigating through a maze of biology data to discover new knowledge. For example, currently in Entrez, when a user starts with a specific protein and wants to find which other proteins it interacts with, she has to manually and painstakingly navigate through hundreds of hyperlinks. This process is very time consuming and error prone. Similarly, if a user wants to query which processes a particular gene participates in, starting from a protein which is translated and transcribed from that gene, she needs to navigate back and forth among protein, DNA, gene, and gene ontology databases.

Fortunately, our approach can make this navigation easy and more efficient because it can use the semantics of the links.

Our link semantics also enable some unique queries and query pathways which currently cannot be accomplished in the Entrez system. Examples of these include determining the domain interactions among proteins and gene regulatory pathway queries. BioLink can automatically help discover domain interactions between two or more proteins using the navigation pathway shown in Figure 5. Starting from a protein id (Protein A), Biolink can find a corresponding DNA sequence using the "*transcribed*" and "*translated*" links for that protein. A specific gene (Gene A) related to this DNA sequence is extracted from the summary of this DNA in the Entrez system. Based on the gene-gene interaction link, BioLink finds a set of genes (Gene B, C, …) interacting with Gene A. It then goes back to the protein database to get a set of proteins (Protein B, C, …) which are translated from Gene B, C, … respectively. This helps establish the set of proteins (Protein B, C, …) that interact with Protein A. Using this interaction information, BioLink generates links between protein A and proteins B, C, … with labels and stores them into the LINK database.

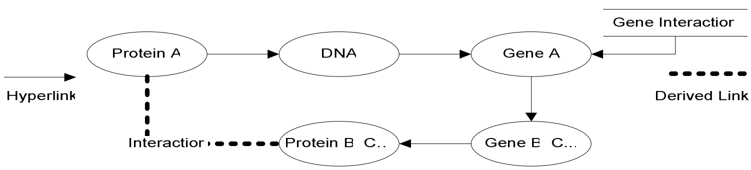



**Fig. 5.** Protein-Protein Interaction

Gene regulation processes are very important and currently cannot be discovered directly from the Entrez system. BioLink makes it possible to generate the gene regulation process using both the link ontology and Gene Ontology (a standard ontology available in life sciences) to capture this important biological process. Given a specific protein (Protein A), BioLink finds a related DNA sequence which is transcribed from that protein and the genes from that DNA sequence. Using Gene Ontology, BioLink then finds other genes that participate in specific regulatory processes with each of these genes and the regulated function in each gene pair. This search process is depicted in Figure 6.

BioLink also helps process popular types of queries such as "which gene is located in which segment of a DNA sequence". This is done using text mining and annotation analysis techniques.

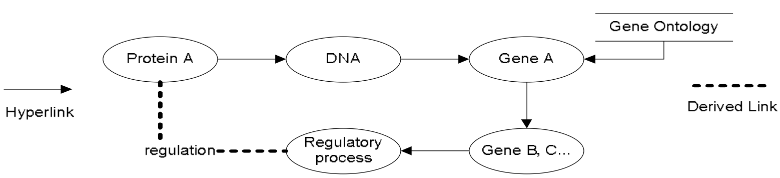



**Fig. 6.** Gene Regulatory Process

Finally, BioLink also facilitates homology searches among sequences and structures. In this type of query, users can set parameters to customize their searches based on similarity scores. For example, given a specific protein (Protein A), users can search for proteins which are homologous to protein A with a threshold similarity score of S, which will exclude sequences with a similarity score of less than the threshold value from the final results.

## 6 Conclusions and Directions for Future Research

Linking biological databases semantically can not only improve the efficiency of queries, but also help discover new knowledge such as protein interactions. In this paper, we develop a comprehensive conceptual model called BioSem to represent biological relationships including biological entities, functions and important processes. Using BioSem, we develop a link ontology and show how to use it to label links between biological databases and then use these to navigate and perform complex queries. With the help of Entrez programming utilities and text mining techniques, we also generate new links between biological entities.

We are in the process of extending this to all of the databases in Entrez and other databases specifically in plant sciences. We also hope to do some more extensive evaluation of the system and the link ontology. We are also working on improving the link generation and navigation techniques. For instance, given an entity, we can get all related links, but some meaningless links should be excluded. In future work, we want to use data mining techniques to inform users which links are more important than others. In addition, for a given query, there is probably more than one navigation path. We are investigating techniques to determine the optimal navigation path for different types of queries.

## References


1. NCBI, `http://www.ncbi.nlm.nih.gov/`
2. DDBJ, `http://www.ddbj.nig.ac.jp/`
3. EBI, `http://www.ebi.ac.uk/`
4. Entrez, `http://www.ncbi.nlm.nih.gov/sites/gquery`
5. Entrez Utility, `http://eutils.ncbi.nlm.nih.gov/entrez/eutils/`
6. UMLS, `http://www.nlm.nih.gov/research/umls/`
7. Ashburner, M., et al.: Gene Ontology: tool for the unification of biology. Nature Genetics 25, 25–29 (2000)
8. Alfarano, C., et al.: The Biomolecular Interaction Network Database and related tools 2005 update. Nucleic Acids Res. 33(Database issue), 418–24 (2005)
9. Bornberg-Bauer, E., Paton, N.W.: Conceptual Data Modelling for Bioinformatics. Briefings In Bioinformatics 3(2), 166–180 (2002)
10. Chen, P.P.: The entity-relationship model—toward a unified view of data. ACM Transactions on Database Systems (TODS) 1, 9–36 (1976)
11. Cornell, M., Paton, N.W., Wu, S., Goble, C.A., Miller, C.J., Kirby, P.: GIMS – A Data Warehouse for Storage and Analysis of Genome Sequence and Functional Data. In: IEEE International Conference on Bioinformatics and Biomedical Engineering, pp. 15–22 (2001)

12. Eilbeck, K., Lewis, S., Mungall, C.J., Yandell, M., Stein, L., Durbin, R., Ashburner, M.: The Sequence Ontology: A tool for the unification of genome annotations. Genome Biology 6(1.5), Article R44 (2005)
13. Heymann, S., Naumann, F., Raschid, L., Rieger, P.: Labeling and Enhancing Life Sciences Links. In: Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference (CSB 2004), pp. 569–570 (2004)
14. Helden, J.V., et al.: Representing and Analysing Molecular and Cellular Function in the Computer. Biol. Chem. 381, 921–935 (2000)
15. Lacroix, Z., Raschid, L., Vidal, M.: Semantic Model to Integrate Biological Resources. In: ICDE Workshops 2006, p. 63 (2006)
16. Lash, A., Lee, W.J., Raschid, L.: A methodology to enhance the semantics of links between PubMed publications and markers in the human genome. Bioinformatics and Bioengineering, 185–192 (2005)
17. Lee, W.J., Raschid, L., Srinivasan, P., Shah, N., Rubin, D., Noy, N.: Using annotations from controlled vocabularies to find meaning associations. In: Cohen-Boulakia, S., Tannen, V. (eds.) DILS 2007. LNCS (LNBI), vol. 4544, pp. 247–263. Springer, Heidelberg (2007)
18. Paton, N.W., Khan, S.A., Hayes, A., Moussouni, F., Brass, A., Eilbeck, K., Goble, C.A., Hubbard, S.J., Oliver, S.G.: Conceptual Modeling of Genomic Information. Bioinformatics 16(6), 548–557 (2000)
19. Ram, S., Wei, W.: Modeling the Semantics of 3D Protein Structures. In: Atzeni, P., Chu, W., Lu, H., Zhou, S., Ling, T.-W. (eds.) ER 2004. LNCS, vol. 3288, pp. 696–708. Springer, Heidelberg (2004)
20. Ram, S., Wei, W.: Understanding Semantic Links among Heterogeneous Biological Data Sources. In: Proceedings of the 15th Workshop on Information Technology and Systems (WITS), Las Vegas, Nevada, December 9-10 (2005)
21. Ram, S., Wei, W.: Semantic Modeling of Biological Sequences. In: Proceedings of the 13th Workshop on Information Technologies and Systems, Seattle, December 12-13, pp. 183–188 (2003)
22. Ram, S., Wei, W.: BIO-LINK: Using Semantics to Link Heterogeneous Biological Databases for Knowledge Discovery. In: Proceedings of 16th Workshop in Information Technology and Systems, Milwaukee, December (2006)
23. Taylor, C.F., Paton, N.W., et al.: A Systematic Approach to Modeling, Capturing, and Disseminating Proteomics Experimental Data. Nature Biotech. 21, 247–254 (2003)

# Integration of Genomic, Proteomic and Biomedical Information on the Semantic Web

Bill Andreopoulos[1,2], Aijun An[2], Xiangji Huang[2], and Dirk Labudde[1]

[1] Biotechnological Centre, Technische Universität Dresden, Germany
williama@biotec.tu-dresden.de
[2] Dept. of Computer Science and Engineering, York University, Toronto, Canada

**Abstract.** Researchers are faced with the challenge of integrating, on the basis of a common semantic web framework, the information on biological processes resulting from genomic and proteomic experimental studies. Researchers would like to integrate the biological processes' roles in larger medical conditions, by taking account of the time dimension. This integration will support automated analysis and reasoning on the semantic web. We address these challenges by proposing the IGIPI framework, standing for "Integrating Gene Interactions and Protein Interactions". IGIPI views different experimental studies as pieces of a puzzle that, if positioned properly, will contribute to a more complete representation of a biological process or medical condition over time. By representing the relative time points of events, this framework represents a biological process based on *how* it might be observed in an experiment over time. IGIPI involves integrating different ontologies and vocabularies, such as GO and UMLS/MeSH. We applied IGIPI to yeast and cancer examples. **Availability:** http://www.cse.yorku.ca/~billa/IGIPI/

## 1 Introduction and Motivation

Biomedical ontologies are often developed in an uncoordinated manner, reflecting hierarchical relations between domain concepts for the purposes of online database annotation and information retrieval. Individual ontologies often are not built for semantic web integration of information from different sources, such as that produced by labs employing different research methods. Ontology development often overlooks the time dimension in biological processes, how to reuse existing ontologies, and consequently ontologies are not interoperable. Ontology interoperability and information integration on the semantic web will support automated analysis, machine processing and reasoning about dispersed online literature, as well as markup of research results online [3]. It is necessary to create a common bioinformatics framework for representing experimental results on biological processes, while integrating the time dimension and previously developed ontologies. Combining these resources is a step towards utilizing their full potential for functional knowledge discovery on the semantic web [2,9].

We present the IGIPI framework for integrating the time-relative information on a biological process, resulting from genomic and proteomic experimental

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 33–42, 2008.

studies. A *biological process* is a network of gene or protein interactions. Integration involves representing the experimental and environmental conditions associated with different studies, under which a biological process may be observed. Moreover, genes' and proteins' contributions under different conditions should be unambiguously represented. Different studies and conditions often suggest conflicting results on a network of gene or protein interactions, highlighting the non-triviality of integration. The contributions of biological processes to a medical condition, such as cancer, can also be represented with IGIPI. A *medical condition* is a condition observed in an organism that is of interest to the biomedical community. IGIPI is based on the notion of 'goals' representing the conditions that need to be satisfied to observe a biological process or medical condition outcome. If an outcome can be observed by two or more different types of experimental studies, such as gene expression studies and two-hybrid studies, then a researcher's aim is to represent the conditions as goals contributing to the overall outcome. A separate representation is developed for each biological process and medical condition, based on an OWL Web Ontology Language specification of IGIPI. Researchers can refine and reuse existing representations for semantic markup of websites.

Researchers need to be able to create a complete picture of the cell by integrating the information resulting from different genomic and proteomic studies [15,13]. One needs to combine the protein interactions observed in two-hybrid studies with the gene interactions observed in synthetic mutant lethality (SML) studies. For this, it is necessary to be able to represent the conditions at different time points under which the protein and gene interactions were observed. Integrating the events observed at the higher cellular level helps to draw more informed conclusions about gene and protein functions and their process involvement, and supports finding knowledge through automated reasoning.

Our objectives in this paper are to provide the ability to represent:

1. a gene/protein inducing a biological process (i.e., a network of gene or protein interactions), while repressing other biological processes.
2. all experimental and environmental conditions under which a biological process was observed.
3. a module of genes/proteins inducing or repressing a biological process.
4. a process consisting of events that changes the module of genes/proteins inducing a biological process, e.g., by attracting more genes to join the module or repelling other genes from the module.
5. the *relative time points* of active modules of genes/proteins and other events in a process.

This paper is organized as follows. Section 2 describes the IGIPI abstractions for biological processes with application on yeast [5]. Section 3 describes extensions of IGIPI for biomedical information with application on cancer. Section 4 discusses analyzing and reasoning with online information. Section 5 puts in the context of related work. Section 6 concludes the paper.

## 2 Integrating Biological Process Information: Timegoals

The IGIPI framework is based on the concept of timegoals. A timegoal is a goal that needs to be satisfied at a specific time interval in an experiment, in order for a biological process to be observed (e.g., a network of protein interactions). Timegoals are goals with no clear-cut criterion for their fulfilment. Instead, a timegoal may only contribute positively or negatively towards achieving another timegoal. By using this logic, a timegoal can be *satisfied* or not. In the IGIPI framework, *satisfying* refers to satisfying at some level a goal or a need, but without necessarily producing the optimal solution.

The IGIPI framework represents information about timegoals using a graphical representation called the *Timegoal Graph (TIG)*. Figure 1 shows an example of a TIG. A TIG records all timegoals representing goals in experiments that, if satisfied, will lead to observing the root biological process. A timegoal is represented as an oval shape, and interdependencies between timegoals are represented as edges. The IGIPI framework supports two types of timegoals: *NFR timegoals* (high level) and *observation timegoals* (low level). The term NFR is derived from the term *non-functional requirement* used in software engineering; in our context an NFR timegoal is a high level goal in an experiment, such as an experimental or environmental condition that needs to be satisfied for observing a biological process, without stating anything about the low level genomic or proteomic events that need to occur. A developer starts constructing a TIG by identifying the top level biological process that is expected to be observed and sketching a root NFR timegoal for it. The root NFR timegoal of a TIG has a value taken from a domain of biological processes. This domain is the *GO Gene Ontology biological process*. The root NFR timegoal is decomposed into timegoals that represent more specific information about how the biological process may be observed.

Timegoals are connected by interdependency links, which show *decompositions* of parent timegoals downwards into more specific offspring timegoals. In some cases the interdendency links are grouped together with an arc; this is referred to as an *AND* contribution of the offspring timegoals towards their parent timegoal, and means that both offspring timegoals must be satisfied to satisfy the parent. In other cases the interdendency links are grouped together with a double arc; this is referred to as an *OR* contribution of the offspring timegoals towards their parent timegoal and means that only one offspring timegoal needs to be satisfied to satisfy the parent. Figure 1 shows that only one of the timegoals for the three types of experimental studies needs to be satisfied, to satisfy the "yeast adaptation to a heat shock" timegoal. When no arc is shown it is an *OR* contribution by default.

The bottom of a TIG consists of the *observation timegoals* representing goals concerning events that need to occur at a low genomic or proteomic level, to satisfy one or more high level NFR timegoals. An observation timegoal is shown as a thick oval shape and represents specific information about a manipulation or an expression of a gene or protein. Observations may be decomposed into more specific observations at a lower level. Observation timegoals make a positive or negative contribution towards satisfying one or more high level NFR timegoals.

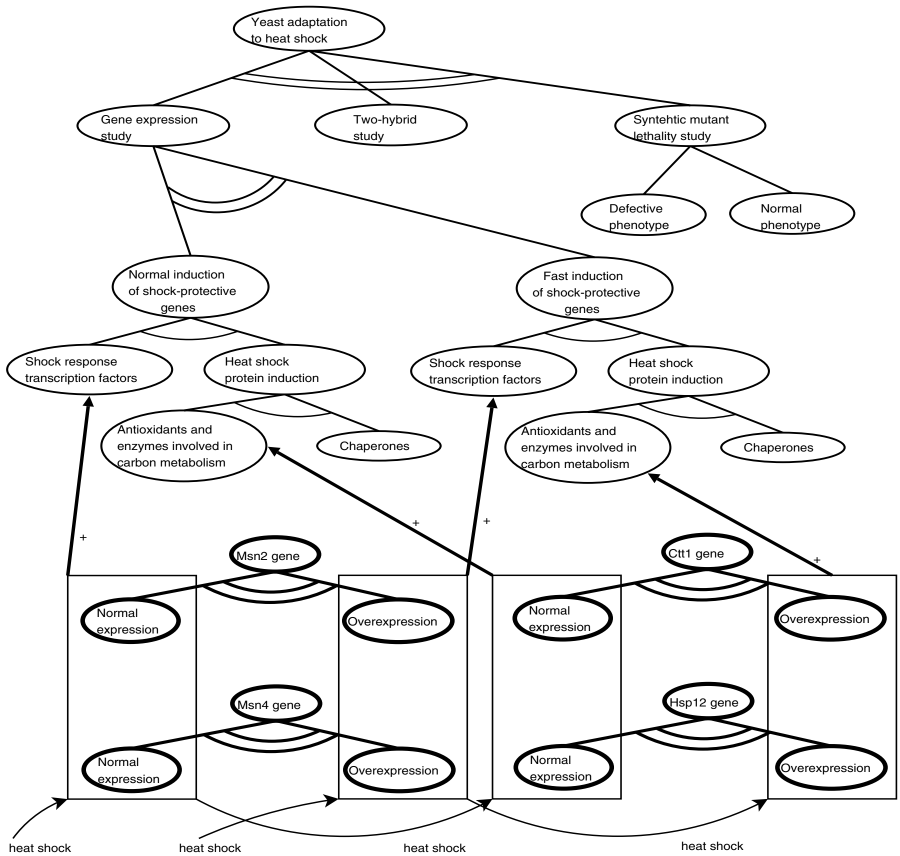



**Fig. 1.** The biological process Timegoal Graph (TIG) for "yeast adaptation to heat shock" [5]. Thick ovals are observation timegoals and thin ovals are NFR timegoals. This figure shows observing the "yeast adaptation to a heat shock" in an experiment as a root NFR timegoal at the top of the TIG. All the different timegoals are arranged hierarchically; a general parent timegoal is decomposed into more specific offspring timegoals at lower levels. An offspring timegoal's time interval is included in the parent timegoal's time interval. To represent the timegoals that need to be satisfied for the "yeast adaptation to a heat shock" to be observed experimentally, the root NFR timegoal is decomposed into the NFR timegoals "gene expression study", "two-hybrid study" and "synthetic mutant lethality study". This means that performing any of these studies leads to observing the yeast's adaptation to a heat shock. The NFR timegoals do not represent information about the low level genomic events that need to occur for the biological process to be observed; this is the purpose of observation timegoals, shown as thick oval shapes. At the bottom is an observation timegoal representing the general goal of observing the Msn2 gene; this timegoal gets decomposed into the timegoals of overexpressing the Msn2 gene and observing the Msn2 gene at its normal expression level. This figure shows a "heat shock" transformation being applied to the overexpressed Msn2 and Msn4 genes, which causes the CTT1 and HSP12 genes to be overexpressed at the next time point.

Figure 1 shows how interdependency links are used to represent an observation timegoal's contribution towards satisfying an NFR timegoal; such a contribution can be positive ('+' or '++') or negative ('-' or '–').

### 2.1 Transformations

The IGIPI framework deals with changes that occur over time in a biological system. It is necessary to represent processes that cause a change in the state of a biological system, both natural processes such as DNA transcription and experimental processes such as mixing [3,15,13]. The IGIPI framework refers to these processes as *transformations*. Transformations are represented as broken lines connecting observation timegoals. The IGIPI framework represents the starting and ending points of a biological transformation as observation timegoals. Timegoals participating in a transformation are observations of proteins or genes' expression levels that contribute towards satisfying a high level biological process. Figure 1 shows that a transformation consists of the participating timegoals, the environmental conditions involved (which may be preconditions for the transformation to occur) and the effects or changes induced by the transformation on the participating timegoals.

### 2.2 Complexes of Genome Components

In a transformation, an event at a time point may involve more than one participating genes or proteins in specific states of expression. The IGIPI framework builds a complete picture of a transformation as it occurs over time, by offering a structural abstraction for representing a group of participants at a time point. This abstraction is called a *complex*. A complex joins several observation timegoals, such as genes or proteins that participate in a transformation simultaneously. Figure 1 shows several examples of gene complexes. When a "normal expression" of Msn2 and a "normal expression" of Msn4 are joined in a complex, together they contribute towards satisfying the "shock response transcription factors" NFR timegoal, thus inducing the process "yeast adaptation to a heat shock".

## 3 Integrating Medical Condition Information

There are web sites with bits and pieces of dispersed information on medical conditions, their symptoms and common side-effects of drugs. Besides building Timegoal Graphs (TIGs) for biological processes, the IGIPI framework can also be used to build TIGs representing information about how medical conditions are manifested. These TIGs can help to integrate biomedical information on the semantic web. We use the term *medical condition timegoals* to distinguish the timegoals of medical condition TIGs from NFR and observation timegoals of biological process TIGs. The root timegoal of a medical condition TIG has a value taken from a domain of medical conditions, such as "ischemic stroke",


**Fig. 2.** The medical condition TIG for "Lung cancer". This figure shows that in some cases it is possible for cells to become drug resistant after chemotherapy. Although it is still not certain how this mechanism works, patients with a turned on gene PKC-epsilon seem to develop drug resistance. This figure shows that for a "protein-induced cell's resistance to chemotherapy drugs (drug-resistant phenotype)" to occur, it is first necessary for "chemotherapy" to occur, combined with the special protein "PKC-epsilon" that is not found in all humans. The symptoms of root timegoal "lung cancer" are grouped under an offspring timegoal named "symptoms". The side-effects of drugs are grouped under an offspring timegoal named "side-effects". Observation timegoals make positive or negative contributions to symptoms and side-effects timegoals that are propagated upwards. This figure shows the decomposition of the Vascular Endothelial Growth Factor (VEGF) into the timegoals "VEGF under Bevacizumab" and "VEGF under Chemotherapy", which represents that the protein is in different states under the influence of Bevacizumab and Chemotherapy. The drug "Avastin" inhibits the VEGF protein. In turn, this contributes negatively to the NFR timegoal "Interaction of VEGF to its receptors" which is getting a negative contribution and thus it is not satisfied. This contributes to the root timegoal of the biological process TIG "Tumor angiogenesis". The biological process "Tumor angiogenesis" contributes to the medical condition TIG that represents information about "Lung cancer". Since "Interaction of VEGF to its receptors" is not satisfied, this contribution is propagated upwards to timegoals "Tumor angiogenesis" and "Lung cancer", neither of which is satisfied either.

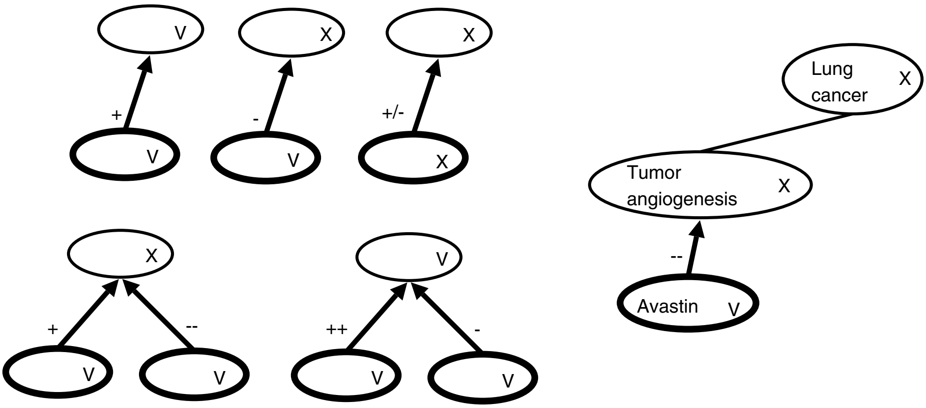



**Fig. 3.** A negative contribution of the "Avastin" drug observation timegoal negatively affects satisfying "Tumor angiogenesis" and "Lung cancer". The symbol '$V'$ on a timegoal means that it is satisfied, while the symbol '$X'$ means that it is not satisfied. "Avastin" is satisfied meaning that this drug is taken by a patient. Contributions from lower timegoals are propagated upwards and contribute towards satisfying higher timegoals. The timegoal "Tumor angiogenesis" contributes to timegoal "Lung cancer", but "Tumor angiogenesis" receives a strong negative contribution from the drug "Avastin" that is taken by a patient; thus "Lung cancer" is not satisfied.

"haemorrhagic stroke", "lung cancer" etc. This domain is the *UMLS Unified Medical Language System* that integrates 100 biomedical vocabularies. Medical condition timegoals can be decomposed into offspring timegoals, which make an $AND/OR$ contribution to a parent timegoal.Medical condition timegoals may also receive positive or negative contributions from the NFR and observation timegoals of biological process TIGs. An NFR or observation timegoal may contribute positively or negatively towards several medical condition timegoals. The contributions are propagated upwards.

Figure 2 shows an example for "lung cancer". The symptoms of a medical condition and the side-effects of drugs are represented as subtrees of the root medical condition timegoal. An observation timegoal is decomposed to represent how it may be observed under the influence of drugs. Figure 3 shows the propagations of contributions for satisfying timegoals.

## 4 Reasoning with Information on the Semantic Web

The ultimate purpose of semantically marking up websites on the basis of IGIPI is to reason on information in a unified manner that could not have been done with traditional online databases. Integrating information on the semantic web means that autonomous agents can scan websites and return to a physician

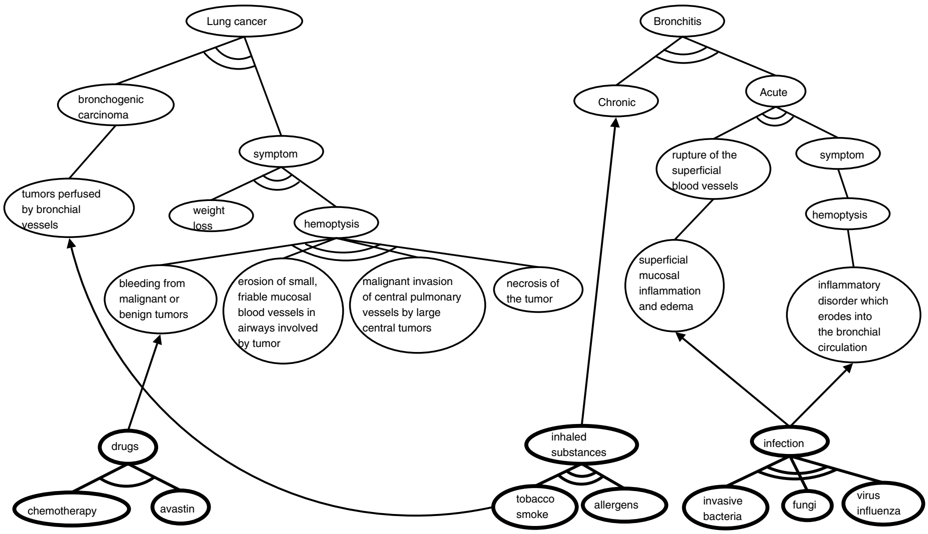



**Fig. 4.** The TIGs for "Lung cancer" and "Bronchitis" both involve the "Hemoptysis" symptom. hemoptysis could be a symptom of lung cancer or bronchitis and in the former case there could be several causes. Serious hemoptysis often occurs in patients with lung cancer when treated with chemotherapy and Avastin. In this case, the incidence of hemoptysis is relatively high in patients receiving chemotherapy and Avastin, as compared to no cases in patients treated with chemotherapy alone. This figure shows that hemoptysis could also be a symptom of bronchitis; in this case it is mild and self-limited. Bronchitis is often a *viral* or *bacterial* disease which follows a cold or infection. A physician who diagnoses hemoptysis as a symptom of bronchitis may be wrong, since the patient could suffer from lung cancer instead. In fact, physicians who work long hours frequently make such errors. How can a physician tell which of all possible conditions holds for a patient with hemoptysis? With a unifying framework to integrate hemoptysis information online for fast lookup and analysis, a physician could make more informed decisions concerning the underlying cause of hemoptysis in a patient.

any semantically annotated information which s/he is not aware of. This involves considering all of the contributions (positive/negative, AND/OR) that are propagated between timegoals in Timegoal Graphs (TIGs). In this section we discuss a case of finding which medical condition is the most probable cause of a symptom observed in a patient.

As an example, suppose a patient is observed with the symptom *hemoptysis*, the act of coughing up blood (Figure 4). Hemoptysis is an important symptom since it frequently reflects serious underlying lung disease. Hemoptysis is often a sign of lung cancer, but it may be caused by different underlying events in lung cancer patients. Hemoptysis also occurs in patients with acute or chronic bronchitis, as well as tuberculosis and pneumonia. Determining the cause(s) of hemoptysis is often not a trivial matter due to the numerous possible causations and their complexities. A physician could use the semantic web to find if the cause of hemoptysis in a patient is likely to be bronchitis or lung cancer or something else.

## 5 Related Work

Individually developed ontologies often support the annotation of online databases for information retrieval purposes. However, they are often not interoperable and do not always allow integration of information derived from different sources and automated reasoning and analysis on the semantic web. Moreover, they do not usually allow the representation of time in modelling. In this section, we outline relevant work of the past two years. In [1], the GeneOntology is recast in OWL to support reasoning. In [4], a top-level ontology of functions is provided. In [6], ontological relations are proposed for enhancing Knowledge Representation and Reasoning. In [7], integrating protein interaction data using the semantic web is discussed. In [8], it is proposed to combine the semantic web with multi-agent systems for integrated access to biomedical information. In [10], integration of neurodegeneration data using RDF is described. In [12], a method exploiting syntactic patterns and the UMLS semantics for aligning biomedical ontologies is proposed. In [16], a Semantic Web system that facilitates cross-database queries and information retrieval in proteomics is proposed. In [17], leveraging the structure of the Semantic Web to enhance information retrieval for proteomics is proposed. In [11], using web ontology language to integrate heterogeneous databases in the neurosciences is discussed. In [14], advancing translational research with the Semantic Web is described.

## 6 Conclusion

We have presented the IGIPI knowledge representation framework, which provides a basis for a powerful use of semantic web technologies (OWL and RDF). Our contributions include the ability to represent the relative time points of events. This approach addresses the problem of biological information integration on the semantic web, through Timegoal Graphs (TIGs) that are built dynamically by the biological community on the basis of the IGIPI framework. This approach supports interoperability between ontologies and vocabularies on the semantic web, including the Gene Ontology, MGED Ontology and UMLS Unified Medical Language System. The Gene Ontology gives values to the root timegoals of biological process Timegoal Graphs (TIGs). The UMLS Unified Medical Language System gives values to the root timegoals of medical condition TIGs. Moreover, this approach supports easy incorporation of new research results.This framework can support automated reasoning on the semantic web, which may allow a physician to relate observed symptoms to a known medical condition, or find likely side-effects of a drug.

## References

1. Aranguren, M.E., Bechhofer, S., Lord, P., Sattler, U., Stevens, R.: Understanding and using the meaning of statements in a bio-ontology: recasting the Gene Ontology in OWL. BMC Bioinformatics 8, 57 (2007)
2. Badea, L., Tilivea, D., Hotaran, A.: Semantic Web Reasoning for Ontology-Based Integration of Resources. In: Ohlbach, H.J., Schaffert, S. (eds.) PPSWR 2004. LNCS, vol. 3208, pp. 61–75. Springer, Heidelberg (2004)
3. Brazhnik, O., Jones, J.F.: Anatomy of data integration. J Biomed Inform 40(3), 252–269 (2007); Epub 2006 (September 24, 2006)
4. Burek, P., et al.: A top-level ontology of functions and its application in the Open Biomedical Ontologies. Bioinformatics 22(14), 66–73 (2006)
5. Cheung, K.H., et al.: YeastHub: a semantic web use case for integrating data in the life sciences domain. Bioinformatics 21 (suppl. 1), 85–96 (2005)
6. Denecke, K.: Enhancing knowledge representations by ontological relations. Stud. Health Technol. Inform. 136, 791–796 (2008)
7. Dhanapalan, L., Chen, J.Y.: A case study of integrating protein interaction data using semantic web technology. Int. J. Bioinform. Res. Appl. 3(3), 286–302 (2007)
8. Garcia-Sanchez, F., et al.: Combining Semantic Web technologies with Multi-Agent Systems for integrated access to biological resources. J. Biomed. Inform. (2008)
9. Koehler, J., Philippi, S., Lange, M.: SEMEDA: ontology based semantic integration of biological databases. Bioinformatics 19(18), 2420–2427 (2003)
10. Lam, H., et al.: AlzPharm: integration of neurodegeneration data using RDF. BMC Bioinformatics 8 (suppl. 3), S4 (2007)
11. Lam, H.Y., et al.: Using web ontology language to integrate heterogeneous databases in the neurosciences. In: AMIA Annu Symp Proc. 2006, pp. 464–468 (2006)
12. Marquet, G., Mosser, J., Burgun, A.: A method exploiting syntactic patterns and the UMLS semantics for aligning biomedical ontologies: the case of OBO disease ontologies. Int J Med Inform. 76 (suppl. 3), 353–361 (2007); Epub 2007 (May 22, 2007)
13. Pasquier, C.: Biological data integration using Semantic Web technologies. Biochimie. 90(4), 584–594 (2008)
14. Ruttenberg, A., et al.: Advancing translational research with the Semantic Web. BMC Bioinformatics 8 (suppl. 3), S2 (2007); Review
15. Rubin, D.L., Shah, N.H., Noy, N.F.: Biomedical ontologies: a functional perspective. Brief Bioinform. 9(1), 75–90 (2008)
16. Smith, A.K., Cheung, K.H., Yip, K.Y., Schultz, M., Gerstein, M.K.: LinkHub: a Semantic Web system that facilitates cross-database queries and information retrieval in proteomics. BMC Bioinformatics 8 (suppl. 3), S5 (2007) Review
17. Smith, A., Cheung, K., Krauthammer, M., Schultz, M., Gerstein, M.: Leveraging the structure of the Semantic Web to enhance information retrieval for proteomics. Bioinformatics 23(22), 3073–3079 (2007); Epub 2007 (October 7, 2007)

# Domain Knowledge Integration and Semantical Quality Management –A Biology Case Study–

Marie-Noelle Terrasse[1], Eric Leclercq[1], Marinette Savonnet[1], Arnaud Da Costa[1], Pierre Naubourg[1], and Magali Roux-Rouquie[2]

[1] LE2I UMR CNRS 5158 - University of Burgundy - France
firstname.lastname@u-bourgogne.fr
[2] LIP6 CNRS-UPMC - 104, avenue du Pt Kennedy - 75016 Paris, France
magali.roux@lip6.fr

**Abstract.** The management of semantical quality is a major challenge in the context of knowledge integration. In this paper, we describe a new approach to constraint management that emphasizes constraint traceability when moving from the semantical level to the operational one.

Our strategy for management of semantical quality is related to a metamodeling-based approach to knowledge integration. We carry out knowledge integration "on the fly" by using transformations applied to models belonging to our metamodeling architecture. The resulting integrated models access available resources through web services whose input and output parameters are guarded by constraints. Integrated metamodels, models, web services and various constraints are produced by ETL (extract-transform-load) operations that are applied both at the model and concept levels. Our metamodeling architectures facilitate scaling up of constraint verification.

**Keywords:** Model Driven Engineering - metamodeling - knowledge integration - biological image annotation.

## 1 Introduction

A major challenge occurring in numerous complex domains is the need of an appropriate balance between global and local semantics of integrated resources. In many domains, including biology, research communities offer portals with web services that can be used to carry out complex computations and data extractions [1]. Such portals generally focus on well-defined sub-domains whose semantics is viewed by domain experts to be unambiguous [2]. These portals represent the first attempt to satisfy the demand for new tools that should improve efficiency of result dissemination. Yet, multiple efforts to generalize the use of mash-up technologies [3] are pushing researchers towards automatically combining portals in order to produce new data and services. In such combinations, quality of the overall results may be quite hard to evaluate.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 43–54, 2008.

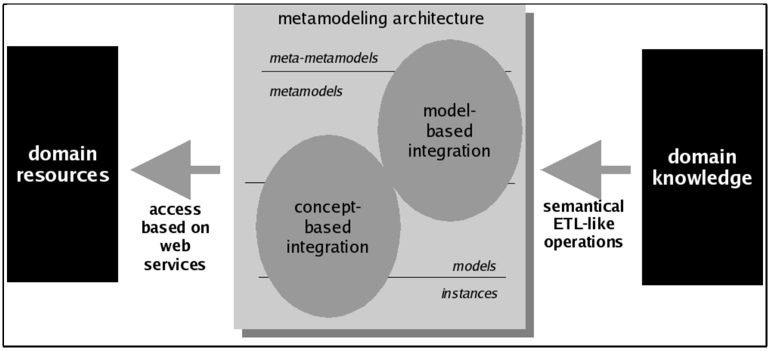



**Fig. 1.** Our metamodeling-based perspective on integration of complex domain knowledge

In this section we first present our strategy (Section 1.1) that partially integrates a domain knowledge by using domain experts knowledge. In Section 1.2 we illustrate two main features of our solution's technical basis, namely metamodeling architectures. The case study on which we present our metamodeling-based strategy for constraints' management is described in Section 1.3.

### 1.1 Our Integration Strategy

As depicted in Figure 1, we construct a metamodeling architecture in order to build a bridge between domain knowledge and domain resources. We use this metamodeling architecture to combine advantages of two major approaches in database integration, namely an approach based on an organization of individual concepts belonging to different systems [4,5,6] and an approach based on the global organization of systems [7,8]. In our proposal, metamodeling architectures describe the structure of domain knowledge so that knowledge integration can be based primarily on global knowledge units that correspond to sub-domains. Furthermore, a concept-level integration deals with fundational concepts.

Our metamodeling architecture contains models derived from domain knowledge and available domain resources. We use semantical ETL-like operations[1] [9] in order to create such a metamodeling architecture. *Semantical extraction* aims at defining a reference universe in terms of consensual resources (e.g., standards, ontologies, underlying theories) and their dependencies. By the end of the semantical extraction, a comprehensive set of resources describing the domain knowledge should be defined. Dependencies between resources are also made explicit. For example: 1) any ontology of the OBO foundry has a referential dependency on the BFO and OBO-RO ontologies; 2) Protégé[2] is dependent on

[1] ETL operations are defined on data-warehouses in order to Extract information that is meaningful for the data-warehouse, to Transform the selected information into the data-warehouse language, and to Load the translated information into the data-warehouse.

[2] Protégé http://protege.stanford.edu/

the OWL and RDF specifications due to compliance requirements. *Semantical transform* aims at translating the chosen consensual resources into a common language that would facilitate knowledge integration. We chose the UML-OCL language for metamodeling architectures and the OWL language for ontologies. In most cases, model transformations [10] will also be available (e.g., from UML to OWL, from OBO to OWL). *Semantical load* aims at organizing consensual resources into a metamodeling architecture in which metamodels achieve wide consensuses while models achieve narrow ones. Semantical load operations also produce diverse modeling artifacts.

### 1.2 Metamodeling Architectures

Our model-based integration process consists in creating metamodeling architectures that express dependencies between consensual descriptions, e.g., standards, good practices guidelines, underlying theories, ontologies. In these metamodeling architectures, *metamodels* describe widely-recognized resources, *reusable models* describe resources recognized by more specialized communities, and *models* describe applications. Ontologies are generally built as more or less general consensual descriptions [11].

Our integration process defines two types of modeling artifacts which are of major importance in our metamodeling architectures, namely semantical variation points and faceted modeling constructs. A *semantical variation point* occurs when two metamodels use incompatible definitions for the same high-level concept (e.g., the concept of time or space). Generally, an ad-hoc integration is carried out at semantical variation points. Such ad-hoc integrations are to be controled by domain experts and the integration must also satisfy exclusion constraints (in order to avoid inconsistencies between integrations carried out by different experts). *Faceted modeling constructs* [12] are used to reduce "modeling-style" variations among representations of the same entity in different contexts. A faceted construct has several facets. An instance of a faceted construct, called a faceted element, possesses a set of such facets. Multiple constraints can be applied to faceted constructs in order to restrict the set of facets that may be actually associated with a faceted element (see Section 2 and Figure 2).

### 1.3 The XenOnt Case Study

In many scientific domains huge collections of images are generated in the context of a single research project. Such images are generally not annotated since this would require large amounts of time and effort from domain experts. In some cases these images can be classified –by using their metadata– into sub-collections of images that are visually similar to each other. We are interested in collections in which images can be described by combinations of simple geometrical shapes. We call such a combination a *pattern*. In order to automatically attribute a pattern to an image, we use semantical information extracted from the image metadata. In order to be able to use such metadata, it is necessary to possess a high-quality domain knowledge description.

The XenOnt application [13] offers an interface for low-cost annotation of images pertinent to development of the olfactory system of the African clawed frog (xenopus laevis). The XenOnt's image collection includes image metadata that make it possible to assign a pattern to each image. Such patterns define all meaningful zones that may be appearing in images. This way domain technicians do not need to have specialized knowledge of the development process of xenopus laevis in order to be able to adjust a pattern to the actual content of an image. Additional measurements can then be made in order to collect quantitative data.

## 2 Semantical ETL Operations

The overall objective of semantical ETL operations is to describe the application's reference universe in terms of knowledge units that correspond to already identified sub-domains. Our metamodeling architecture (metamodels, reusable models and models) is devoted to the global integration of knowledge while various modeling artifacts (e.g., faceted constructs) serve to facilitate concept-level integration [12]. Semantical ETL operations for the XenOnt application are presented in more detail below. An overview of the overall semantical ETL is then offered in Figure 3.

*Semantical extraction* is organized into three steps. The preliminary and core steps are model-based extraction processes while the third step is a concept-based process.

The *preliminary semantical extraction* aims at defining a set of dependencies between resources. These dependencies must be meaningful in the application domain. For our XenOnt application, this sample set contains three dependencies. The first dependency is a relation of thematic coupling between resources (e.g., the FuGE standard [14] and the OBO FBbi ontology have scientific experiment as a common theme). The second dependency is a reference relation between the XenOnt application and external vocabularies and standards (e.g., the XenOnt application references the OBO XAO ontology of xenopus laevis). The third dependency is a compliance relation between a specification and an implementation. Such compliance can be partial, e.g., when a single component of the implementation is compliant, while the specification is irrelevant for other components of the implementation.

The *core semantical extraction* aims at selecting and organizing resources that are of interest for the application. Resources are organized according to types of dependency chosen during the preliminary semantical extraction. This core semantical extraction is carried out at the model level, which means that all dependencies need to be defined between resources. Seven resources that are relevant to the XenOnt application are given below. In order to facilitate the long-term evolution of the XenOnt application, it was decided very early in the development process to use OBO ontologies and OWL-RDF languages as much as possible. Furthermore, domain experts have identified six resources that are needed for domain-specific annotation of images. The first resource contains two basic ontologies, namely BFO and OBO-RO, that are used to describe

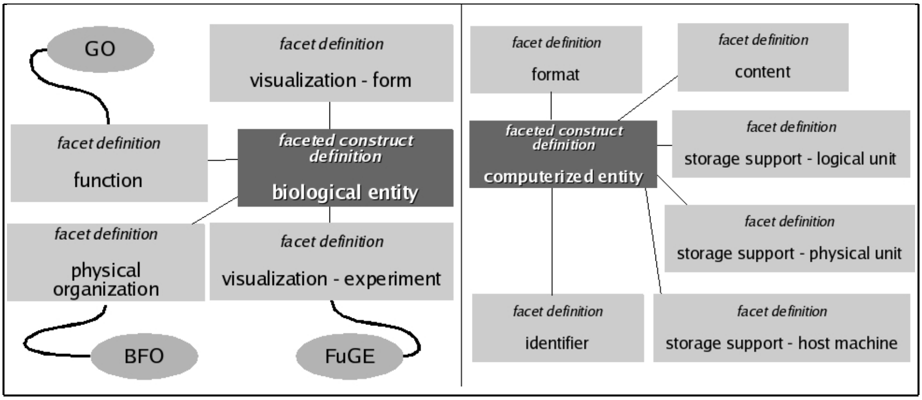



**Fig. 2.** Faceted constructs (biological entities and computerized entities). Faceted constructs are represented by dark-grey rectangles, facets are represented by light-grey rectangles. External resources are represented by ovals. Dependencies of a facet on an external reference are represented by thick curves.

spatio-temporality and to establish a typology of relations between concepts. The second resource is the FuGE standard's common part that contains "*components used to develop models for high-throughput or data intensive experimental processes. This includes support classes for auditing, security, references to external information, measurements and protocols*". The third resource is the OBO "xenopus anatomy and development" ontology, denoted by XAO, that offers "*a structured controlled vocabulary of the anatomy and development of the African clawed frog*". This ontology is used to annotate the research target (i.e., an organ at a given development stage). The fourth resource is the OBO "biomedical imaging methods" ontology, denoted by FBbi, that offers "*a structured controlled vocabulary of sample preparation, visualization and imaging methods used in biomedical research*". This ontology is used to annotate experimental conditions in order to facilitate data comparison and experimental results' reproducibility. The sixth resource contains the domain specific literature and thesauruses.

The PubMed and the SUDOC[3] publication portals serve as example resources for the XenOnt application.

The *concept-level semantical extraction* aims at identifying fundamental concepts, i.e., concepts that are likely to be used by most of the domain applications, even though their uses in different applications may be different. Domain experts have identified two fundamental concepts that are defined in terms of faceted constructs [12]. The first faceted construct is that of biological entity (see the left side of Figure 2). A biological entity is a domain-related faceted construct whose facets are 1) its physical organization (described in terms of BFO concepts such

[3] SUDOC: *The Sudoc catalog provides bibliographic information on all types of documents held by French academic libraries, including PhD theses*, http://www.sudoc.abes.fr/LNG=EN/.

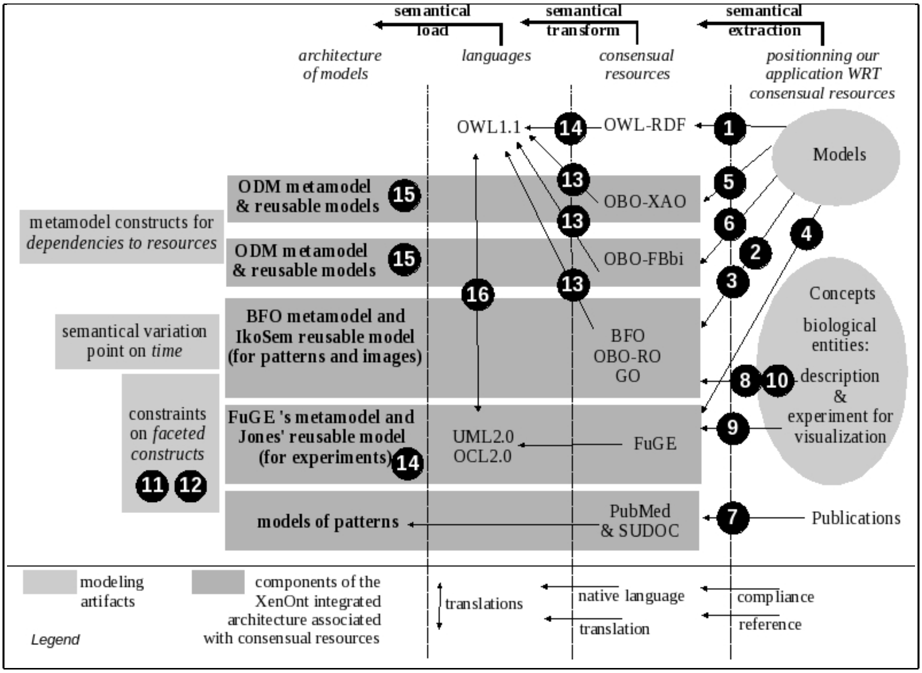



**Fig. 3.** Overview of semantical ETL operations (white numbers in black bullets represent Rules 1 to 18). Columns from right to left represent: right-most colum) consensual resources of the XenOnt application domain; second from right column) the seven resources extracted for the XenOnt application; third from right column) translation of extracted resources into the XenOnt target languages; fourth from right column) metamodeling components; left-most column) modeling artifacts.

as substance, part, and aggregate), 2) visualization facets such as its forms and experimental parameters (described in terms of FuGE concepts), 3) its biological function in terms of the Gene Ontology (GO). A biological entity thus has dependency relations with the BFO and GO ontologies and with the FuGE standard. The following constraint is applied to our biological faceted entity: "a faceted element with a visualization-form facet must possess a physical organization facet and a visualization-experiment facet". The second faceted construct is that of computerized entity (see the right side of Figure 2). It is a data management related faceted construct whose facets are 1) its format, 2) its storage support facets including a host-machine (computer or dedicated equipment), physical and logical units, 3) its identifier, 4) its content. The following constraint is applied to our computerized faceted entity: "each faceted element must possess at least one storage support facet (i.e., storage support host-machine, storage support physical and logical unit)".

We obtain the following rules that are applied to the XenOnt application:

**Rule 1-** *XenOnt has a compliance relation with OWL-RDF*
**Rule 2-** *XenOnt has a reference relation with BFO*
**Rule 3-** *XenOnt has a reference relation with OBO-RO*
**Rule 4-** *XenOnt has a reference relation with FuGE*
**Rule 5-** *XenOnt has a reference relation with OBO-XAO*
**Rule 6-** *XenOnt has a reference relation with OBO-FBbi*
**Rule 7-** *XenOnt has a reference relation with PubMed*
**Rule 8-** *Biological entities have a compliance relation with BFO*
**Rule 9-** *Biological entities have a compliance relation with FuGE*
**Rule 10-** *Biological entities have a compliance relation with GO*
**Rule 11-** *Biological entities have three mandatory facets, namely visualization-form, physical organization, and visualization-experiment*
**Rule 12-** *Computerized entities have at least one storage support facet.*

*Semantical transform* for the XenOnt application has a high semantical accuracy since various authors have been interested in matchings between the OBO language, UML-OCL and OWL [15,16,17,18]. As depicted in Figure 3, the OWL translations of all selected ontologies are used within the XenOnt application. Translations from the OBO format to the OWL format are to be carried out under all constraints formulated by Golbreich & al. [15]. The FuGE specification has been originally written in UML. Yet, in [12] we discussed the possibility of working under alternative recommendations, namely those of Jones & al. [14]. We thus need to formulate the corresponding rules in terms of OCL constraints attached to FuGE's UML diagrams. The XenOnt application itself generates an ontological description in terms of OWL, within OWL DL limits.

We obtain the following rules:

**Rule 13-** *OBO to OWL transformation must satisfy Golbreich & al. requirements*
**Rule 14-** *FuGE must be used together with Jones & al.'s recommandations*
**Rule 15-** *XenOnt's ontological description must satisfy OWL DL limitations.*

*Semantical load* creates a metamodeling architecture which is depicted in Figure 4. Due to the importance given to dependencies between a target application and consensual resources, a dedicated metamodeling construct is defined for modeling of such dependencies. Since we plan to translate descriptions from UML to OWL, we introduce an entity-relation representation of ontologies, namely the OMG's Ontology Definition Metamodel (ODM) [16], into our metamodeling architecture. It is thus necessary to make sure that 1) our metamodeling architecture is compliant with the OMG's organization of metamodels including UML, RDF, and OWL metamodels[4], and 2) UML modeling constructs that have been excluded from the ODM metamodel[5] are not used in UML models of ontologies.

[4] [16] pages 53–54.
[5] [16] pages 69–71.

The FuGE specifications and Jones' guidelines are introduced into the XenOnt architecture in the form of metamodels and reusable models, respectively. BFO is introduced as a metamodel derived from both the UML and ODM metamodels. Since the meaning of time in BFO and UML might be different, a semantical variation point on time is introduced.

The XenOnt project is based on our earlier work, namely the IkoSem project [19] for building image databases from collections of images, oriented towards semantical and physical CBIR (content-based image retrieval). An IkoSem metamodel was intended for modeling images that contain simple geometrical shapes. This metamodel contains various constructs for modeling of images, image parts, relations between image parts (including spatial relations), and attributes to be associated with image parts and images.

In the XenOnt metamodel, which itself is derived from Ikosem and BFO metamodels, we offer constructs for 1) patterns and zones of patterns (which are derived from images and image parts), 2) ontological concepts that must be linked to consensual resources by using our dependency construct, 3) a new construct for attributes in order to model the mandatory association of attributes with ontological concepts.

We obtain the following rules:

**Rule 16-** *A metamodeling construct for dependencies is needed*

**Rule 17-** *UML models of ontologies must not contain modeling constructs that have been excluded by the ODM specification*

**Rule 18-** *A semantical variation point on time (associated with BFO metamodel) is needed.*

## 3 Modeling the XenOnt Application

Translations from UML to OWL and from OWL to UML are likely to be carried out on the fly during the XenOnt application's lifecycle. We thus have to make sure that we use non-contradictory UML and OWL semantics. We chose to apply Evermann & al.'s guidelines [20]. Moreover, OCL constraints that cannot be translated into OWL must not lost.

We first define a reusable model, *XenOnt-rm*, for annotation of images by using metadata-determined patterns. We then derive the XenOnt application's model from this reusable model. An additional rule must be defined in order to select OBO ontologies that will be linked to attributes of pattern zones (i.e., XAO) and image attributes (i.e., XAO and FBbi).

We obtain the following constraints:

**Rule 19-** *UML to OWL transformations must comply with the ODM metamodel and with Evermann & al. guidelines*

**Rule 20-** *Pattern attributes can only be associated with XAO ontology concepts.*

When semantical ETL operations have been carried out, many rules (Rules 1 to 4 and 8 to 18) will be taken into account within components of the metamodeling architecture itself. One of the remaining rules (Rule 19) needs to be

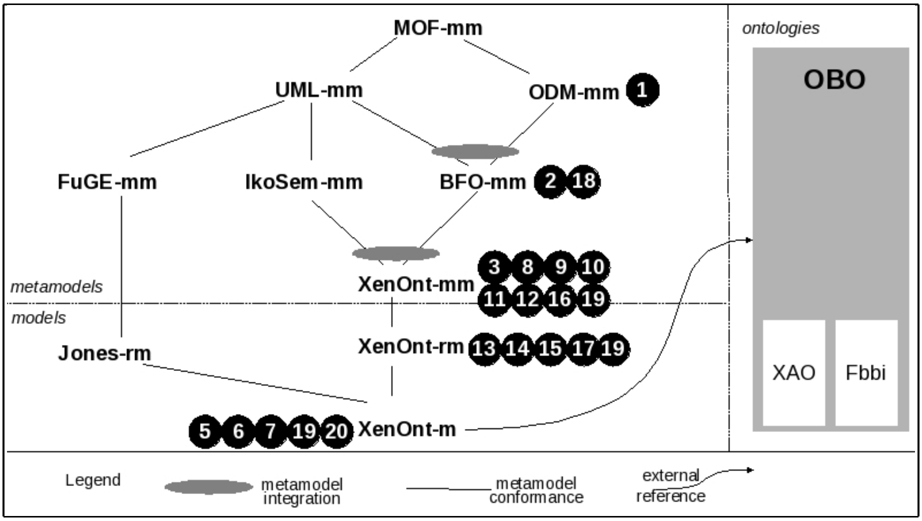



**Fig. 4.** Metamodeling architecture for the XenOnt application (white numbers in black bullets represent Rules 1 to 20)

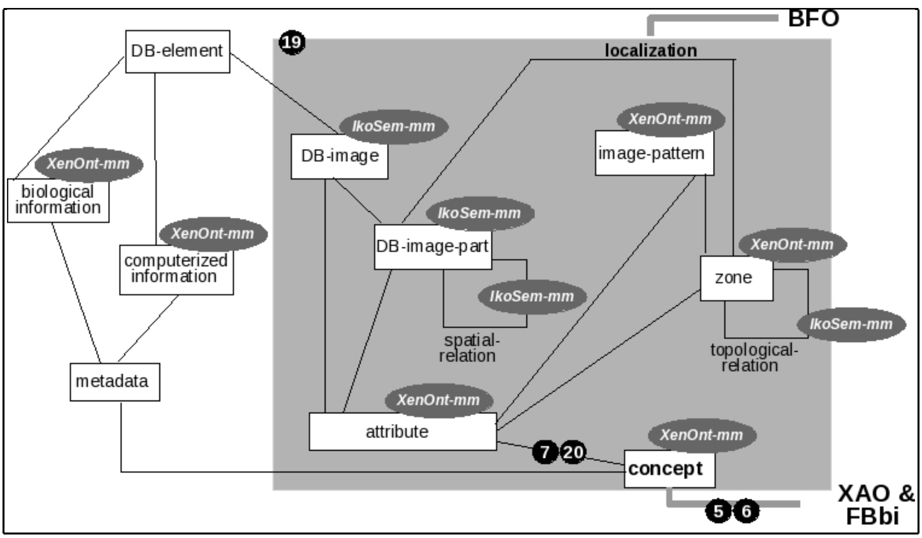



**Fig. 5.** A perspective model of the XenOnt application (oval dark shapes indicate XenOnt-mm and IkoSem-mm metamodeling constructs, respectively; white numbers in black bullets represent Rules 1 to 20). The grey background rectangle indicates the scope of Rule 19. Thick grey lines represent external references.

taken into account at the metamodel, reusable model, and model levels. Other rules (i.e., Rules 5, 6, 7, 19 20) are pushed down to the model level.

## 4 Conclusion

In this paper, we propose to support engineering of complex applications by using a metamodeling architecture. Such an architecture will play the role of an intermediate description between knowledge models that are used by domain experts and those of domain resources. We propose to generalize the above approach to other domains that offer multiple heterogeneous and evolving web-enabled resources. In the context of research-related domains, our metamodeling architecture is to be built as described below:

1. Definition of a collection of resources that are of interest for the application's users. Such resources are to be defined by domain experts. Since consensual models and ontologies form the core of domain resources, we propose the following guidelines for collecting resources:
   - Choice of fundamental ontologies and metamodels such as BFO, OBO-RO, ODM that provide a description framework for high-level concepts. Various discussions that can facilitate decisions at this stage have been published, e.g., comparison of upper-level ontologies [21], comparison of UML-based and ontology-based approaches of domain modeling [20]. Such fundamental resources should be used for any application domain.
   - Choice of widely recognized resources of the application domain (e.g., FuGe standard, Gene Ontology, and EBI portals for the biological domain). This choice should be well-balanced between resources that are actually usable by domain communities and major standardization efforts.
   - Choice of application-specific resources.
2. Definition of a comprehensive set of relations between the above resources and the application. The set of relations should include at least reference, compliance and thematic coupling of resources. Various other relations can be used in application domains whose resources are thightly tied together. For example, in the biological domain two complementary relations could be used for species coupling and technological space coupling.
3. Insertion of all resources into a metamodeling architecture coupled with ontologies. Various modeling constructs can be used in order to build an integrated architecture. The architecture's upper level should contain metamodels related to the ontological and UML modeling paradigms (e.g., MOF, UML and ODM; or MOF, UML and Bunge metamodel [22]) and to fundamental ontologies (e.g., BFO, Bunge). Domain resources should occupy an intermediate position (metamodels and reusable models) and be organized using selected domain-specific relations. The architecture's lower level should contain application-related models.

4. Definition of the application's model within the above architecture. The application model should contain various model transformations Model transformations on ontologies should be defined according to Noy's classification [24].

Multiple rules (e.g., dependency relations, constraints) are formulated during the above four stages. These rules are expressed, whenever it is possible, by our modeling constructs which allows to automatically enforce the rules within models (e.g., Lu & al. proposed a comprehensive test of enforcement of Evermann's rules in ArgoUML [23]). Rules that can not be included in metamodels are expressed within the application model and enforced in the application's implementation.

Various model transformations may need to be carried out in order to build the application software and to control its evolution. For example Noy [24] described the state-of-the art of frameworks dedicated to ontology based integration that distinguishes between various levels of ontology transformations (e.g., ontology-ontology, instance-instance). As much as possible, constraints to be applied to model transformations (e.g., Evermann's rules [20]) should be formulated on reusable models in order to improve constraint verification reuse. Such model and ontology transformations must comply with the domain semantics defined by our metamodeling architecture.

By using such an architecture, we expect a major improvement of the quality of applications.

## References

1. Labarga, A., Valentin, F., Anderson, M., Lopez, R.: Web services at the european bioinformatics institute. Nucleic Acids Research Advance 35, W6–W11 (2007)
2. Crampin, E., Halstead, M., Nielsen, P.H.P., Noble, D., Smith, N., Tawhai, M.: Computational physiology and the physiome project. Experimental Physiology 89(1), 1–26 (2002)
3. Li: Enterprise interoperability – research roadmap – version 4.0. Technical report, European Community, Information Society Technologies (2006), http://cordis.europa.eu/ist/ict-ent-net/ei-roadmap_en.htm
4. Atzeni, P., Torlone, R.: Managing heterogeneous data models in a graph-theoretic framework. In: Proceedings of the 3rd International Conference on Information and Knowledge Management, CIKM 1994, Gaithersburg, Maryland, USA (1994)
5. Jeusfeld, M., Johnen, U.: An executable meta model for re-engineering of database schemas. In: Proceedings of the $13^{th}$ International Conference on Entity-relationship Approach, Manchester, UK (1994)
6. Lou, J.: Data model description and translation using the meta-model SOME. Ph.D thesis, EPFL, Lausanne, Switzerland (1997)
7. Falkenberg, E.D., Han Oei, J.: Meta model hierarchies from an object-role modeling perspective. In: Proceedings of the $1^{st}$ International Conference on Object-Role Modeling, ORM-1, Magnetic Island, Australia (1994)
8. Frederiks, P., ter Hofstede, A., Lippe, E.: A unifying framework for conceptual data modelling concepts. Information and Software Technology 39(1), 15–25 (1997)

9. Vassiliadis, P., Simitsis, A., Skiadopoulos, S.: Conceptual modeling for etl processes. In: DOLAP, pp. 14–21 (2002)
10. Czarnecki, K., Helsen, S.: Classification of model transformation approaches. In: Proceedings of the OOPSLA Workshop on Generative Techniques in the Context of Model-Driven Architecture (2003)
11. Bao, J., Hu, Z., Caragea, D., Reecy, J., Honavar, V.: A tool for collaborative construction of large biological ontologies (2006)
12. Terrasse, M.N., Roux, M.: Metamodeling architectures for complex data integration in biology. International Journal of Biomedical Engineering and Technology Special issue Warehousing and Mining Complex Data: Applications to Biology, Medicine, Behavior, Health and Environment 3(2) (to appear, 2009)
13. Da Costa, A., Leclercq, E., Gaudin, A., Gascuel, J., Terrasse, M.: Organizing metadata into models and ontologies for lowering annotation costs of a biological image database. Studies in Computational Intelligence (2008) ISSN : 1860-949X
14. Jones, A., Miller, M., Spellman, P., Pizarro, A.: Fuge: Funtionnal genomics experiment model specification. FuGE Working Group, Recommandation (draft) (2007)
15. Golbreich, C., Horrocks, I.: The OBO to OWL mapping, GO to OWL 1.1! In: Proceedings of the 3rd Int. Workshop on OWL: Experiences and Directions (OWLED 2007) (2007)
16. ODM: Ontology definition metamodel (August 2005), http://www.omg.org/docs/ad/05-08-01.pdf
17. Motik, B., Horrocks, I., Sattler, U.: Bridging the gap between owl and relational databases (2007)
18. Patel-Schneider, P., Horrocks, I.: A comparison of two modelling paradigms in the semantic web. ACM, New York (2006)
19. Besson, L., Savelli, J., Leclercq, E., Terrasse, M.N.: Ikosem's generic model for syntactical and semantical image description: the core component of a framework for image database engineering. In: Proceedings of the 10th International Workshop on Multimedia Information Systems, Maryland, USA (2004)
20. Evermann, J., Wand, Y.: Ontology based object-oriented domain modelling: fundamental concepts. Requirements Engineering 10, 146–160 (2005)
21. Mascardi, V., Corde, V., Rosso, P.: Comparison of upper-level ontologies. Technical Report DISI-TR-06-2, DISI (Univ. degli Studi di Genova) Italy (2006)
22. Evermann, J.: A UML and OWL description of Bunge's upper-level ontology model. Software and Systems Modeling (2008), doi:10.1007/s10270-008-0082-3
23. Lu, S., Parsons, J.: Enforcing Ontological Rules in UML-Based Conceptual Modeling: Principles and Implementation. In: Halpin, T., Siau, K., Krogstie, J. (eds.) Proceedings of the Workshop on Evaluating Modeling Methods for Systems Analysis and Design (EMMSAD 2005), held in conjunctiun with the 17th Conference on Advanced Information Systems (CAiSE 2005), Porto, Portugal, EU, FEUP, Porto, Portugal, EU, pp. 451–462 (2005)
24. Noy, N.F.: Semantic integration: a survey of ontology-based approaches. SIGMOD Rec. 33(4), 65–70 (2004)

# Towards a Scientific Model Management System

Fabio Porto[1], José Antônio de Macedo[1], Javier Sanchez Tamargo[1], Yuanjian Wang Zufferey[1], Vânia P. Vidal[2], and Stefano Spaccapietra[1]

[1] EPFL-IC – Database Laboratory
Lausanne, Switzerland
{fabio.porto,jose.macedo,javiersancheztamargo, yuanjian.wangzufferey,stefano.spaccapietra}@epfl.ch
[2] UFC - Department of Computing
Ceará, Fortaleza, Brazil
vvidal@lia.ufc.br

**Abstract.** Computational models of biological systems aim at accurately simulating in vivo phenomena. They have become a very powerful tool enabling scientists to study complex behavior. A side effect of their success unfortunately exists and is observed as an increasing difficulty in managing data, metadata and a myriad of programs and tools used and produced during a research task. In this work we aim at supporting scientists during a research endeavour by using Scientific Models as a main guiding element for describing, searching and running computational models, as well as managing the corresponding results. We assume a data-oriented perspective for scientific model representation materialized into a data model with which users describe scientific models and corresponding computational models, and a query language with which a scientist specifies simulation queries. The model is grounded in XML and tightly related to domain ontologies, which provide formal domain descriptions and uniform terminology. Scientists may search for scientific models and run simulations that automatically invoke the underlying programs on provided inputs. The results of a simulation may generate complex data that can be queried in the context of the scientific model. Higher-level models can be specified through views that export a unified representation of underlying scientific models.

## 1 Introduction

Multiple large-scale scientific projects are expected to produce a never before observed amount of experimental and simulation data. Current technology for data management is clearly unable to cope with the needs to store, index, search, analyse, process and integrate such massive, complex and very often-heterogeneous data. Additionally, the very aim of scientific exploration requires a different perspective on data management, one that can cope with expected knowledge evolution and worldwide collaboration.

A promising approach is based on the concept of scientific models [1], which are formal representations synthesizing the understanding about a studied phenomenon, entity or process. It allows scientists to simulate the behaviour of the real world and compare it against experimental results.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 55–65, 2008.

From a data perspective, scientific models encompass all the information used and produced during a scientific exploration. It is contextualized by its scientific domain, bibliographic references, a description of the observed phenomenon and, in some cases, some mathematical formulae and provenance information. A computational model, in this paper, refers to a realisation of a scientific model through computational resources, software and hardware. Basically, a scientist or engineer creates a computational implementation that simulates the behaviour of the studied phenomenon and its interaction with interfacing environment, according specifications in the scientific model.

Transforming a scientific model into a computational one can be a complex task, if at all possible, but once realised allows scientists to confront simulation against experimental results, leading eventually to an accurate representation of the studied phenomenon. This process known as model tuning includes both modifying programs (i.e. changing the behaviour) and fitting input parameter and set-up values (i.e. specifying initial simulation state). In many cases, when experimental data are available, regression analysis, provide formal procedures for model fixing [2].

It is a fact that the complexity of specifying and running a computational model and managing all the resources produced during a scientific endeavour deviates the attention off the phenomenon being investigated to implementation concerns [3]. State of the art approaches to computational simulation rescue to scientific workflows languages, such as SCUFL and BPEL, to specify workflows and to their corresponding running environments [4,5, 6,7] for models evaluation. For small scale non data intensive simulations, scientist may choose to use tailored environments like MATLAB [8] and Neuron [9].

Whilst some of these have attracted a large user community [10], they fail to provide an integrated scientific model environment as proposed in this paper, in which the studied phenomenon, the derived models and data therein produced are integratedly managed.

In this work we aim at supporting scientists in specifying, running, analysing and sharing scientific models and model's data. To this end we sketch a scientific model management system architecture and detail a data model and query language for specifying scientific models and running simulations. We assume a data-oriented perspective for scientific model representation, materialized into a data model with which users describe scientific models and derived computational models, and a query language with which a scientist specifies simulation queries. The model is grounded in XML and tightly related to domain ontologies, which provide formal domain descriptions and uniform terminology. Scientists may search for scientific models and run simulations that automatically invoke the underlying programs on provided inputs. The results of a simulation may generate complex data that can be queried in the context of the scientific model. Higher-level models can be specified through views that export a unified representation of underlying scientific models.

The remaining of this paper is structured as follows. Section 2 gives a brief introduction to the system architecture. Next, section 3 introduces a running example extracted from the neuroscience domain. Section 4 describes the data model and section 5 presents the simulation query language. Finally, section 6 concludes discussing our achievements so far and future works.

## 2 Scientific Model Management System

A scientific model management system (SMMS) supports scientists in designing, creating, searching and running scientific models, and in managing the results of simulations and analysis. Figure 1 depicts the main system functions.

The system functions are structured into four layers. A user layer provides the interface for scientists to create and edit elements of the model and to request system services, such as running simulations, querying and reasoning. Users may query scientific model meta-data as well as simulation results.

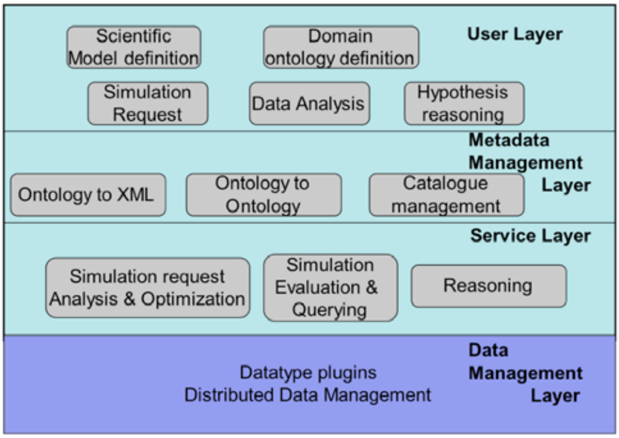



**Fig. 1.** Scientific Model Management System Architecture

The metadata management layer stores scientific model metadata and supports metadata management services. In this work, scientific model metadata is based on a set of ontologies that guarantees uniform terminology among models and scientists. A transformation and selection service allows scientists to map ontology fragments to XML trees, which are then used in data model elements description. The catalog service manages metadata about scientific model and data, as well as supporting information such as ontologies, views (see section 5.5) and transformation rules.

The service layer supports simulation evaluation, querying of simulation results and reasoning. We have extended the query processing system CoDIMS [11] to cope with simulation queries evaluation.

Finally, a data management layer supports distributed scientific models management and wrappers implementing complex datatypes, offering access to simulation results, such as graphs and time series.

In this paper, the details of the architecture are not further explored. Instead, the following sections present the backbone of the system in the form of a data model and simulation language. Similarly, the details regarding ontology managing, transformation and alignment are left to future work.

## 3 A Neuroscience Scientific Model

In this section we introduce a running example taken from scientific models developed for the neuroscience domain. According to Kandel [12], “the task of neural science is to explain behavior in terms of the activities of the brain”. Numerous computational neuroscience groups investigate scientific models that aim at explaining such behavior. A classical example is the axon membrane action potential model proposed by Hodgkin and Huxley (HH) [13] that describes how action potentials traverse the cell membrane. An action potential is a pulse-like wave of voltage that can traverse certain types of cell membranes, such as the membrane of the axon of a neuron. The model quantitatively computes the membrane potential between the interior and the extracellular liquid, based on the flow of ions of sodium ($Na^+$), potassium ($K^+$) and a leakage one representing all other channel types. The mathematical equation proposed by HH is presented below:

$$I = m^3\ h\ g_{Na}\ (E - E_{Na}) + n^4\ g_K\ (E - E_K) + g_L\ (E - E_L)\ (1),$$

where $g_i$, i={Na, K, l}, are function of time dependent variables, representing the membrane conductance, and $E_i$ model the equilibrium potential for each ion channel, while E is the membrane potential. Finally, n, m and h are parameters controlling the probability of the sodium or potassium gates to be opened. The total ionic current across the membrane of the cell is modeled by the variable I.

The action potential model defined by HH simulates the variation on voltage in the cell when applied to a neuron compartment. From a conceptual point of view, one can represent a single neuron model with its various compartments and the membrane behavior given by the HH model by the ontology in Figure 2. The Hodgkin-Huxley class models the corresponding model with the input and output parameters conforming to equation (1).

From the scientific model specification, a computational neuroscientist will conceive programs that implement the behavior defined by equation 1. Next, when running a simulation, the program receives input values for the parameters identified in (1) and produces the total ionic current across the membrane (variable I in (1)).

In the next section, the HH model is used as a running example illustrating the scientific model data model elements.

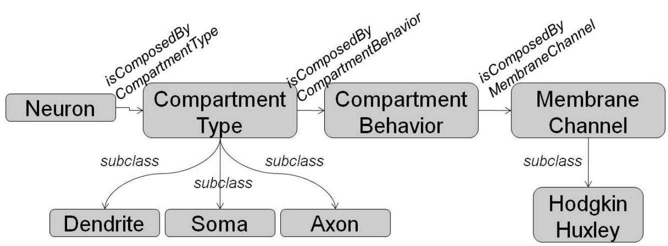



**Fig. 2.** Single Neuron with Hodgkin-Huxley model domain ontology

# 4 The Scientific Model Data Model

The backbone of the SMMS is a data oriented semantically based data model and simulation language. During a scientific exploration, a scientist registers data and metadata describing the scientific and corresponding computational models. Scientific model metadata provides provenance, contextual and descriptive information aiding on model search and querying. Similarly, computational model metadata is used as the basis for the automatic evaluation of simulations and serves as context for the input and output data. Thus, the data model identifies the following main composing elements: the scientific model, the computational model and simulations. The following subsections detail each one of these layers.

## 4.1 The Scientific Model Layer

The first layer describes a Scientific Model (SM). It accompanies the first step in an experiment or simulation specification, in which scientists describe the research problem being pursued. It considers the scientific domain, setting the context to the problem, and the formal problem specification in the form of mathematical formulae, if adequate. A formal description of a SM is given in (1) and includes: a resource definition *R*, a list of bibliographic references (B), possibly images (I), and annotations (A). *R* is a resource description defined as a tuple <LSID, D>. LSID is an unified identifier for the life science specified as urn:lsid:authority.org:namespace:object:revision, where *urn:lsid* is the LSID protocol identifying label, *authority.org* is a DNS reference, and the remaining field names are self explanatory [14]. *D* is a free text resource description. We use domain ontologies to formally describe the domain covered by the scientific model ($O_{SMD}$) and to explain the mathematical formulae ($O_{MF}$). The LSID identification attribute hooks the scientific model to its computational models and simulations.

Thus, a scientific model is defined as a 6-tuple $\mathbf{SM}=\langle \boldsymbol{R}, \boldsymbol{O_{SMD}}, \boldsymbol{O_{MF}}, \boldsymbol{B}, \boldsymbol{I}, \boldsymbol{A} \rangle$ (1).

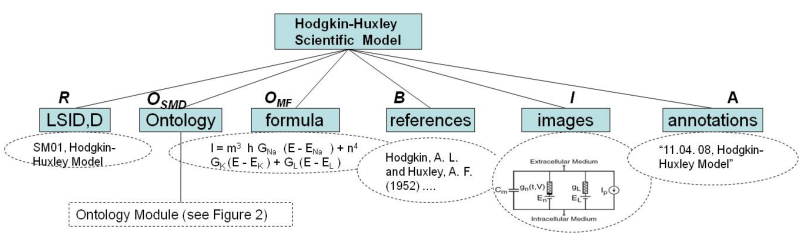



**Fig. 3.** The HH scientific model representation

The HH model, presented in section 3, can be depicted as a scientific model. Its data view is illustrated in Figure 3. It provides metadata to support basic search queries over a scientific model database and reasoning capabilities on the theory specified by the scientific model ontology.

### 4.2 The Computational Model Layer

A Computational Model (CM) realises a Scientific Model through a computational implementation. From the point the view of the research task, a scientist develops software that reproduces the behavior specified by the scientific model. The objective of the computational element is to describe its components and provide sufficient metadata to allow automatic instantiation of a CM when requested by a simulation query (see section 5.4). The computational model may comprehend a series of orchestrated programs whose scheduling is hidden by an exposed façade entry point. Indeed, a computational model is the basis for a simulation query that provides values matching CM input requirements to produce the simulation output.

A CM specification comprehends two ontologies: environment and domain. The former describes the execution environment associated to the CM, including: programming language specification, programming environment parameters, documentation, input and output parameters, initialization procedure, and executing programs. The set of input parameters expected by a CM can be divided into two groups: set-up and input parameters. The CM set-up parameters include state information that should hold during various simulations. Set-up parameters may also introduce run-time control values, such as frequency of output graph update etc. The set-up parameter are referred to in the data model as $I_S = \{i_1, i_2, \ldots, i_n\}$, where each instance $i_j = \langle v, P \rangle$ of $I_s$ corresponds to a value $v$ and its corresponding parameter $P$ in Parameters.

Additionally, for each computational model, an *outputWrapper* specifies methods for obtaining its results. Given the different formats and procedures used by generic programs when producing output, each CM shall indicate the *outputWrapper* class that knows how to capture its output and return it to the system to feed an eventual pipeline.

The CM domain ontology contextualizes CM parameters according to the $O_{SMD}$ ontology. Contextualization is obtained by associating each parameter to its semantic meaning as an instance of a ontology concept. In fact, we propose a transformation of the domain ontology into a XML serialization tree, in which nodes correspond to ontology concepts, and edges, from parent to children are either ontology roles, having the role domain being the parent node in XML, or is-a relationships. Thus, as shown in Figure 4, the output variable "I" is mapped to the XML path "Neuron/Axon/Hodgkin-Huxley/I", which in turn corresponds to a path in the ontology specified in DL-SWRL [15] as:

Neuron(?x) ^ isComposedByCompartmentType(?x, ?y) ^ Axon(?y) ^
isComposedByCompartmentBehavior(?y, ?z) ^ CompartmentBehavior(?z) ^
isComposedByMembraneChannel(?z, ?a) ^ HodgkinHuxley(?a) ^
hasTotalIonicCurrent(?a, ?I) → TotalIonicCurrent (?I)

A CM model is formally defined as a 7-tuple $\mathbf{CM} = \langle \boldsymbol{R}, \boldsymbol{LSID_{SM}}, \boldsymbol{XO_E}, \boldsymbol{XO_D}, \mathbf{M_i}, \mathbf{M_o}, \boldsymbol{A} \rangle$ (2).

In (2), $R$ is the CM resource identification; $LSID_{SM}$ is a reference to the associated scientific model, $XO_E$ and $XO_D$, are the XML serializations of the environment ontology, of XML type Environment, and of the domain ontology, of XML type Neuron, respectively. In addition, $M_i$ ($M_o$) defines mappings between the underlying program input (output) parameters and corresponding domain ontology properties (XML tree leave node). Finally, *A* corresponds to annotations identifying authoring information.

Figure 4 illustrates the representation of a computational model implementing the scientific model SM01.

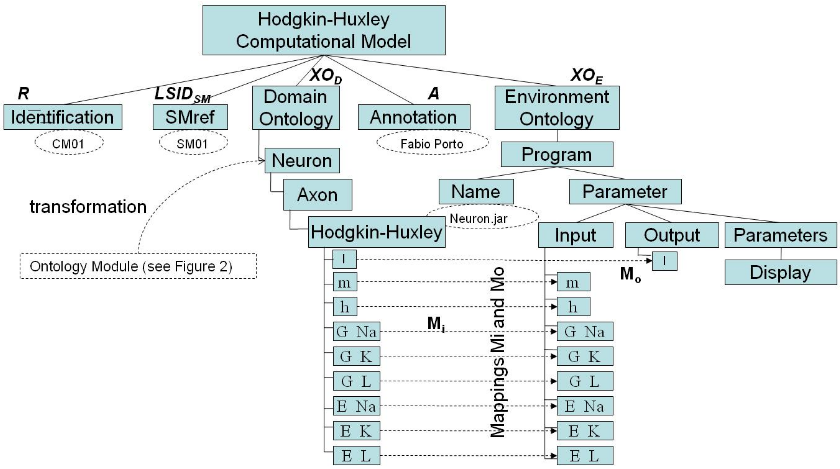



**Fig. 4.** A Hodgkin-Huxley Computational Model

### 4.3 Simulation Layer

The third layer in the scientific model data model specifies a simulation query (S). Whilst the two first layers present metadata about a scientific model and its computational implementation, a simulation query combines CMs into an evaluation. A simulation query specification provides the initial state of the simulation through its input parameter values and a particular set-up. Section 5 will describe the simulation query language.

## 5 Simulation Language

Syntactically, a simulation query is specified as an expression comprising a head and a body. The body is a boolean expression composed of a conjunction of simulation query predicates, whereas the head lists variables containing the expected simulation results, necessarily appearing in one of the simulation predicates in the body. We start by presenting the syntax and semantics of simulation query predicates.

### 5.1 Simulation Predicate

A simulation query predicate is specified as:

$$\mathbf{S_i((V_i,W_i)\ ;\ (X_i', X_o')\ ;\ (I_i,O_i)\ ;\ \mathit{I}_S)}\ (3).$$

In (3), $S_i$ labels the simulation query predicate according to the CM resource identification. In order to easy references to the computational model input and output values, two sets of variables are defined, respectively, $V_i$ and $W_i$. These variables refer to values provided as input or produced as output when running the underlying CM. Indeed, users interface with simulations queries by providing input parameter and set-up values, and by receiving back the output values computed as a function of these inputs. The set of input and output parameters' values are provided by the XML documents $X_i$' and $X_o$', respectively, whereas $I_S$ represents simulation set-up parameters. Note that the associated CM definition specifies the schemas for $X_i$' and $X_o$'. For example, using the CM example in Figure 4, the $X_i$' document can be obtained from the result of the XPath expression "/CM/DomainOntology" over the HH CM XML definition, and by filling its leaf nodes with the input values. Thus, /Neuros/Axon/Hogking-Huxley/m = 0,1, illustrates a possible value assignment for the input parameter *m*. Finally, the mappings $I_i(O_o)$ define the correspondence between the input-output variables in $V_i$ and $W_i$ and the input-output parameter values in $X_i$' and $X_o$'.

**Definition 1.** correspondence assertions in $I_i$ and $O_i$ are specified as $x = *Path*, where $x is a variable in $\{V_i \cup W_i\}$ and *Path* is an XPath [16] expression pointing to a data element in $X_k$', k={i,o}, whose child node is either an input parameter value or an output value.

Having described the syntax of individual simulation predicates, the semantics of body expressions can be announced. Initially, the semantics of a single simulation predicate is exposed followed by one for complete body expressions.

### 5.2 Semantics of a Single Simulation Predicate

A single simulation predicate returns a boolean value of its evaluation according to the definition in Def 2 below with respect to the its syntax in (3) and the CM specification in (2).

**Definition 2.** A simulation predicate $S_i$ is assumed to be *true* iff given a $X_i$' holding the set of input parameter values to the program implementing the corresponding CM, according to $M_i$, it exists a $X_o$' whose leaf values are produced by the evaluation of the referred program and that is built from the mappings in $M_o$.

### 5.3 Semantics of the Body of a Simulation Expression

Once the semantics of a single simulation predicate is specified as a boolean expression, more elaborate conjunctive expressions can be composed to form the body of a simulation. The semantics of a conjunction of simulation predicates in the body of a simulation is defined in Def 3.

**Definition 3.** Given a conjunction of simulation predicates $s = s_1 \wedge s_2 \wedge \ldots \wedge s_n$, $s$ is considered to hold *true* if the conjunctive expression on the right evaluates to true. Moreover, if more than one simulation predicates $s_i$ and $s_j$ in $s$, refer to the same variable, for instance $x, then they share a single associated value. In addition, the shared variable must hold a single binding to a value, either provided as input or produced as output by an underlying program computation.

Note that the restriction regarding sharing variables among simulation predicates leads to data-dependency relationships, in which the simulation predicate holding the associated value to the shared variable shall precede in evaluation order the remaining simulation predicates sharing that particular variable. Moreover, variable sharing introduces a particular mode of value assignment to data elements in $X_i$‘, replacing that of the node corresponding to its associated path.

Finally, given a body that evaluates *true*, then the head of the simulation identifies the variables in the simulation predicates whose values are returned as the simulation outputs, such that if K is the set of variables in the head then

$K \subseteq (V_i \cup W_i)$, for $1 \leq i \leq n$, with $n$ being the number of the simulation predicates in the body.

### 5.4 A Simulation Query

A simulation query combines the head and its body into a simulation clause as illustrated in (4), according to Def 1, 2 and 3.

$$\begin{aligned} S(K) = S_1((V_1,W_1) ; (X_{i1}', X_{o1}') ; (I_1,O_1) ; I_{S1}) \wedge \\ S_2((V_2,W_2) ; (X_{i2}', X_{o2}') ; (I_2,O_2) ; I_{S2}) \wedge \quad (4) \\ \ldots\ldots \qquad \wedge \\ S_n((V_n,W_n) ; (X_{in}', X_{on}') ; (I_n,O_n) ; I_{Sn}) \end{aligned}$$

An example of a simulation query is depicted in Figure 5. This particular query returns the total ionic current across the membrane ($I) according to the parameters values specified in the input document $HHCM01_I$. As discussed before, the user must provide a mapping from each query variable to the corresponding data element of the domain ontology XML serialization document. In this example, the input and output XML documents, **$X_i$’** and **$X_o$’**, are illustrated by documents $HHCM01_I$ and $HHCM01_O$, respectively, both of type Neuron.

```
S($I, $Z) = CM01(($m,$h,$G_Na,$G_K,$G_L,$E_Na,$E_K,$E_L,$I);
                  (HHCM01_I, HHCM01_O);
                  ($m = /Neuron/Axon/ Hodgkin-Huxley/m,
                   ....[1],
                  $I = /Neuron/Axon/ Hodgkin-Huxley/I)) ∧
            CM02[2] (($I , $Z); (ACM02_I, ACM02_O);
                  ($Z=/Analysis/result))
```

**Fig. 5.** Simulation query example

### 5.5 Simulation View

One may want to register a simulation so that it can be re-executed later on or included in a more complex simulation. A registered simulation is called a view, borrowing the term from database literature [17], as it provides users with an external

[1] The remaining mappings are not shown due to lack of space.
[2] The CM02 computational model has purposely not been described.

perspective of a simulation through the set of input parameter values that configure the computational models taking part into the simulation view. In addition, a simulation view establishes correspondences between the exported parameters and the ones specified on each simulation predicate taking part in the body of the simulation describing the view. In (5) a simulation view is depicted:

$$\begin{aligned} S_v((V, W) ; (X_{iv}', X_{ov}') ; (I_v, O_v) ; (I_{Sv}, M_s)) = \\ S_1((V_1, W_1) ; (X_{i1}', X_{o1}') ; (I_1, O_1) ; I_{S1}) \wedge \\ S_2((V_2, W_2) ; (X_{i2}', X_{o2}') ; (I_2, O_2) ; I_{S2}) \wedge \\ \ldots \wedge \\ S_n((V_n, W_n) ; (X_{in}', X_{on}') ; (I_n, O_n) ; I_{Sn}). \end{aligned} \quad (5)$$

The body of the simulation view reflects the one in ordinary simulations, expressing conjunction of simulation predicates. The difference appears in the head of the formulae. Indeed, the latter exports an integrated view of the simulation predicates' input and output parameters appearing in the body of the formulae and specified in $X_{ik}$'and $X_{ok}$', $1 \leq k \leq n$. The two sets of correspondences, $I_k$ and $O_k$, map the external view in $\{X_{iv}', X_{ov}'\}$ to the corresponding parameters in the simulation predicates in the body, $\{X_{ik}', X_{ok}'\}$. Thus, a correspondence assertion is expressed as $S_v$.*path/dataelement* ≡ $S_i$.*path/dataelement*, where path is an XPath expression. In the same line as the input/output parameters, $I_{Sv}$ expresses the uniform view of set-up parameter values appearing in the body of the formulae and $M_s$ asserts the correspondences between the set-up data elements in $I_{Sv}$ and those in the body.

## 6 Conclusion

Managing *in silico* simulations have become a major challenge for eScience applications. As science increasingly depends on computational resources to aid solving extremely complex questions, it becomes paramount to offer scientist mechanisms to manage the wealth of knowledge produced during a scientific endeavor. This paper presents initial results aiming to contribute to this idea. We propose a data centric semantic based data model with which scientists represent scientific models and associated computational models. Furthermore, the data model supports specifying and running simulations as a function of computational models over inputs. The intention of this work is to provide a basic backbone from which more complex services can be developed as their needs come. We have developed a first prototype system that implements the data model and the simulation query language on top of the CoDIMS system. A first scientific model management system architecture is proposed with a set of minimal services that scientists may expect from such an environment.

There are many future works already being explored. Indeed, the investigation of scientific hypothesis may introduce uncertainty to the data model. Another problem is related to the format and semantic mismatch between computational models data in a simulation. We are interested in studying how the results on heterogeneous databases and ontologies can be applied to the scenario discussed here. Finally, the integration of simulation results with knowledge in ontologies and the exploration of their content as complex data structures within a query language is also a subject of future research.

## References


[1] Hunter, J.: Scientific Models – A User-oriented Approach to the Integration of Scientific Data and Digital Libraries. In: VALA 2006, Melbourne (February 2006)
[2] Jaqaman, K., Danuser, G.: Linking data to models: data regression. Nature Reviews, Molecular Cell Biology V(7), 813–819 (2006)
[3] Silvert, W.: Modelling as a discipline. International Journal General Systems V30(3), 1–22 (2000)
[4] Oinn, T., Greenwood, M., Addis, M.: Taverna: Lessons in creating a workflow environment for the life sciences. Concurrence Computation: Pract. Exper., 1–7 (2000)
[5] Akram, A., Meredith, D., Allan, R.: Evaluation of BPEL for Scientific Workflows. Cluster Computing and the Grid, CCGRID V1(16-19) (2006)
[6] `http://www.gridworkflow.org/snips/gridworkflow/space/XScufl`
[7] Altintas, I., Berkley, C., Jaeger, E., Jones, M., Ludascher, B., Kepler, M.: An Extensible System for Design and Execution of Scientific Workflows. In: SSDBM (2004)
[8] MATLAB (last access, 24/06/2008), `http://en.wikipedia.org/wiki/Matlab`
[9] Neuron (last access, 24/06/2008), `http://www.neuron.yale.edu`
[10] Roure, D., Goble, C., Stevens, R.: Designing the myExperiment Virtual Research Environment for the Social Sharing of Workflows. In: e-Science 2007 - Third IEEE Int. Conf. on e-Science and Grid Computing, Bangalore, India, December 10-13, 2007, pp. 603–610 (2007)
[11] Porto, F., Tajmouati, O., Silva, V., Schulze, B., Ayres, F.: QEF - Supporting Complex Query Applications. In: CCGRID 2007, Rio de Janeiro, Brazil, pp. 846–851 (2007)
[12] Kandel, E., Schwarts, J., Jessel, T.: Principles of NeuroScience, 4th edn. McGraw-Hill, New York (2000)
[13] Hodgkin, A., Huxley, A.: A quantitative description of ion currents and its applications to conduction and excitation in nerve membranes. J. Physiol (Lond.) 117, 500–544 (1952)
[14] (last accessed, 26/04/2008), `http://lsids.sourceforge.net/`
[15] Grosof, B., Horrocks, I., Volz, R., Decker, S.: Description Logic Programs: Combining Logic Programs with Description Logic. In: Proc. WWW 2003, Budapest (May 2003)
[16] (last accessed, 26/04/2008), `http://www.w3.org/TR/xpath`
[17] Elmasri, R., Navathe, S.: Fundamentals of Database Systems, 2nd edn. Benjamin/Cummings (1994)

# Preface to ECDM 2008

Fabio Grandi

University of Bologna, Italy

Change is a fundamental but sometimes neglected aspect of information and database systems. The management of evolution and change and the ability for database, information and knowledge-based systems to deal with change is an essential component in developing and maintaining truly useful systems. Many approaches to handling evolution and change have been proposed in various areas of data management and this forum seeks to bring together researchers and practitioners from both more established areas and those from emerging areas to look at this issue. The fifth ECDM workshop (the first four ECDM workshops were held with ER'99 in Paris, France, with ER'02 in Tampere, Finland, with ER'04 in Shanghai, China, and with ER'06 in Tucson, Arizona) deals with the manner in which change can be handled, and the semantics of evolving data and data structure in computer based systems, including databases, data warehouses and Web-based information systems.

With respect to the main ER conference, the ECDM workshop series aim at stressing the evolutionary aspects involved in conceptual modelling and in development and implementation of systems, ranging from the modelling of information dynamics to the dynamics of the modelling process itself. Another explicit aim of ECDM workshops is to bring together scientists and practitioners interested in evolution and change aspects in different research fields and, thus, often belonging to completely separate communities. As a result, such an interaction could become tighter and cross-fertilization more useful in the context of a collaborative workshop like ECDM.

Following the acceptance of the workshop proposal by the ER 2008 organizing committee, an international and highly qualified program committee was assembled from research centers worldwide. As a result of the call for papers, the program committee received 8 submissions from 6 countries and after rigorous refereeing 3 high quality papers were eventually chosen for presentation at the workshop and publication in these proceedings, with a 37.5% acceptance ratio. Furthermore, the workshop program is enriched by a keynote speech given by Carlo Zaniolo, leading specialist in the temporal XML field but also enduring friend of the ECDM workshop series.

I would like to express my thanks to the program committee members and the additional external referees for their timely expertise in reviewing, the authors for submitting their papers, the ER 2008 organizers for their support.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, p. 66, 2008.

# Time Versus Standards: A Tale of Temporal Databases

Carlo Zaniolo

Computer Science Department
University of California, Los Angeles, CA
zaniolo@cs.ucla.edu

**Abstract.** Because of the importance of time in Information Systems, there is a wide consensus on the need for having temporal data types and temporal queries supported as part of database standards. Actually, in the mid 90s, a forward-looking group of database researchers proposed an ambitious temporal extension of SQL-2 named TSQL2. Unfortunately, this proposal was ahead of its time and was not well-received by standard committees and DBMS vendors represented in those committees. Since then, standardization groups have not spent much effort on temporal issues, and, hence, it is natural to assume that no progress was made, and the gap between database standards and temporal applications remains as wide as it was in the mid-90s. This would represent very bad news for temporal application developers who suffer from inadequacies of database standards and do not see any relief in the near future. Fortunately, time conquers all and has also transformed standards, making them more supportive of time-oriented applications, in ways that are not fully recognized even by those who, motivated by different goals, pushed for the introduction of the new features.

The first significant rapprochement between standards and temporal applications occurred with the introduction of XML and XQuery, which, respectively, support a temporally grouped representation, and provide a powerful and extensible query language. These enable users to effectively model temporal information and pose complex historical queries without requiring changes in the standards. For all the merits of XML, its limited performance and scalability, combined with unwavering loyalty of users to relational data models and query languages, suggests that SQL will remain the crucial testbed for a successful rendezvous of standards and time. In this context, the SQL:2003 standards mark a significant step forward because of their OLAP Functions, which entail efficient expressions of temporal aggregates, and other important temporal functions, including coalescing. An even more exciting development is represented by the recently proposed new SQL standards for finding patterns in sequences of rows. By employing these Kleene-closure based constructs on suitable representations, users can then express very interesting temporal queries in SQL. Moreover, the power and flexibility of these extensions overcome the rigid division between event-based temporal data models and state-based models—a very desirable unification since, in the real world, a salary-increase event and the state of being at the new salary level are viewed as two facets of the same history.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, p. 67, 2008.

# Modeling Transformations between Versions of a Temporal Data Warehouse

Johann Eder and Karl Wiggisser

Alps-Adria University Klagenfurt, Inst. of Informatics-Systems
{eder,wiggisser}@isys.uni-klu.ac.at

**Abstract.** Data warehouses are oftentimes in charge of supporting the decision finding processes in companies and public administration. To fulfil these tasks the systems model parts of real world regarding the respective application domain. But the world is changing and in such an evolving environment data warehouses can only cope with their tasks if they can be kept consistent with the real world. For that purpose they have to be able to deal with modifications in the data schema and their influence on the data values. This paper focusses on the modeling of the data transformation. We identified six different types of transformation operations. Based on the semantic analysis of these operations we present a matrix based representation for data transformation which enables us to specify the transformation results and a more efficient graph based representation, which furthermore offers potential for optimization.

**Keywords:** Data Warehouse Maintenance, Temporal Data Warehouses, Data Transformation.

## 1 Introduction

Data warehouses (DWH) as a building block in a decision support system depend on the reliability and accuracy of contained data. Most DWH do not only comprise current snapshot data but also historical data to enable analysis over a long period of time. And, as we live in a changing world, one criterion for reliability and accuracy of the results of such long period queries is their comparability.

Consider an analysis for the inhabitants of the European Union for the years 1986–2006. Such a query leads to several semantic problems: What about countries that were not part of the EU for the whole period? Should inhabitants of these countries be counted, and where to get the data from? Former West and East Germany reunited in 1990, how should such a case be treated? In 2004 East European countries joined the union. Some of them did not even exist until the early nineties. How should these countries be included in the result? What "version" of the countries should be considered? For instance, Eurostat [1] lets the user choose the "Geopolitical Entity" (i.e. "version" of the EU) to use for the analysis. We could give numerous similar examples, but the purpose is clear: In a changing environment, DWH must evolve to be able to execute their tasks.

The most popular architecture for DWH are *multidimensional databases*, where *transactional data* – i.e. cell values – are described in terms of *master*

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 68–77, 2008.

*data* or *structure data* – i.e. dimensions, categories (also called the *schema*), and dimension members (also called the *instances*). Whereas DWH systems are well prepared for changes in the transactional data, they are, surprisingly, not able to deal with changes of master data very well.

To overcome these limitations, methods of *Data Warehouse Maintenance* have to be implemented. Data warehouse maintenance can be defined as follows: "The process and methodology of performing changes on the schema and instance level to represent changes in the data warehouse's application domain or requirements is called *Data Warehouse Maintenance. Data Warehouse Evolution* is a form of data warehouse maintenance where only the newest data warehouse state is available. *Data Warehouse Versioning* is a form of data warehouse maintenance where all past versions of the data warehouse are kept available." [2] One aspect of DWH maintenance – i.e. the notion of different versions of a DWH and change identification between these versions – has been handled in previous work [3,4]. This paper focuses on dealing with the impact of structure changes on the cell data. The following issues have to be considered: (i) *Semantics of structure changes and impact on cell data*: There are several types of structure changes, ranging from simple updates to complex operations including many structure elements. The semantics of all these operations and their impact on the cell data have to be defined. (ii) *Models for data transformation*: To handle the data transformation induced by structure changes it is necessary to define models that allow to specify the desired results of the transformation and provide an efficient representation of the transformation. For this purpose, this paper addresses the syntactic and semantic definition of data transformations. We present a matrix based representation which eases the specification of the transformations and a more efficient graph based representation which offers potential for optimization.

## 2 The COMET Temporal Data Warehouse

The COMET Metamodel for Temporal Data Warehouses [3] is based on the multidimensional datamodel. Thus, it is possible to represent all elements of a a DWH cube's structure definition (dimensions, categories, dimension members, user defined attributes, ...) with COMET. Furthermore, it is also possible to integrate cell data into such a cube representation.

Each of the structure elements and relations between cube elements get assigned a timestamp defining their validity. If an element is changed, a new version of this element is created, so a version history is recorded. Elements with the same end of validity build a so called *structure version*, which represents a version of the cube structure how it was valid at a certain point in time. Between different versions of a dimension member so called *transformation functions* are defined, which transform the assigned cell values between different structure versions. Such a transformation function is defined as $MapF(SV_i, SV_j, DM_i, DM_j, w)$, where $SV_i$ and $SV_j$ are the two structure versions between which is transformed, $DM_i$ and $DM_j$ are the dimension members between which is transformed ($DM_i \in SV_i$ and $DM_j \in SV_j$, $DM_i$ and $DM_j$ belong to the same dimension). $w$ is the so called transformation weight, i.e. the factor which is used to transform the cell values assigned to $DM_i$ by multiplying the cell value with the transformation value. Following this definition, it is clear that COMET only allows to deal with numeric data, which seems reasonable, as most analytic cubes contain numeric data. This paper presents a semantic analysis of the transformation functions' impact and their efficient representation.

### 2.1 Related Work

There are several approaches addressing the problem of changing DWH structures, which considerably differ in their capabilities. The approaches can be classified by their supported type of maintenance (evolution or versioning) and the level on which they support maintenance (schema only, instances only, schema and instances) [2]. Kimball [5] was the first presenting means for DWH evolution on instances. Evolution on the schema level is, among others, proposed by Quix [6]. Evolution on schema and instances is introduced by, for instance, Vaisman and Mendelzon [7]. Body et al. [8] present an approach for versioning on the instance level. Versioning on schema level is presented by Golfarelli et al. [9]. Versioning for schema and instances is supported by Malinowski and Zimányi [10]. Kaas et al. [11] present an approach for schema evolution on star and snowflake schemas based on a relational data store. Besides these scientific approaches, there are also two commercial approaches by SAP Inc.[12] and Kalido [13], but their capabilities are rather limited, as they both only act on the instance level.

## 3 Data Transformation in a Temporal Data Warehouse

Besides managing the structural changes in a DWH, also the impact of these changes on the cell data have to be taken care of. Changes of dimension members often imply a change in the semantics of the respective members and with such a semantic update also the semantics of the associated cell values may change. To achieve semantically correct and comparable query results it may be necessary to adjust cell values to the new situation. This is done with the help of so called *Data Transformation Operations*. In this section we describe these operations and present two different methods for modeling them. The matrix representation is intended for easy specification of the operations' results, whereas the graph representation is more efficient and offers potential for optimization.

### 3.1 On the Equality and Identity of Dimension Members

The versioning process deals with versions of DWH structure elements. In this context we have to define the two aspects of *Equality* and *Identity*, especially for dimension members. Two dimension members are said to be *equal* iff all their attributes (e.g. name, comment, ...) are equal. Two dimension members stemming from different structure versions are called *member versions* of the same member iff they represent different versions of the *same real world entity*, i.e. the *member's identity* is preserved. There are some aspects of equality and identity which are important to note: (i) Equal members in different structure versions are not necessarily member versions of the same dimension member (e.g. an employee left the company and another person having the same name is hired instead). (ii) Two member versions of the same dimension member are not necessarily equal, but there could have been a change (e.g. the phone number of an employee is changed). (iii) There can be two or more dimension members that are equal within one structure version, although this may not make much sense (e.g. two employees with the same name and phone number). (iv) There cannot be two or more member versions of the same dimension member within one structure version. (v) The identity of a dimension member cannot be automatically decided, but this needs human interaction.

### 3.2 Data Transformation Operations and Their Semantics

There are six transformation operations. Each of them consists of one or more transformation functions, as defined in the COMET Metamodel.

**Update.** The update operation changes a source member version $a$ of structure version $u$ into a destination member version $a'$ of structure version $u+1$, where $a$ and $a'$ are versions of the same (in terms of identity) dimension member. An update consists of one transformation function between $a$ and $a'$ with transformation value $v$. All cell values associated with the updated member version are multiplied with the transformation factor, giving a new cell value associated with the destination member version. An update can, for instance, describe the change of the currency for the member "Turnover" from ATS to EUR. The transformation value is 0.07267, which is the official exchange rate.

**Split.** The split operation divides one source member version $a$ from structure version $u$ into a set of destination member versions $\{a_1, \ldots, a_n\}$ of structure version $u+1$, where $a$ and $a_i \in \{a_1, \ldots, a_n\}$ are not versions of the same dimension member. A split consists of $n$ transformation functions, one between $a$ and each $a_i$ with a transformation value $v_i$. All cell values associated with the respective member versions are multiplied with the respective transformation factor. Each multiplication result is then a new cell value associated with one of the new member versions. A split can, for instance, describe the division of a department $X$ into three new departments $X_1$, $X_2$ and $X_3$. Department $X$ itself stops existing. None of the new departments can be seen as direct successor (in the sense of, for instance, legal successor) of $X$. The transformation values could express the fractions in which $X$ is divided.

**Merge.** The merge combines a set of source member versions $\{a_1, \ldots, a_n\}$ of structure version $u$ into a single destination member version $a$ of structure version $u+1$, where $a_i \in \{a_1, \ldots, a_n\}$ and $a$ are not versions of the same dimension member. A merge consists of $n$ transformation functions, one between each $a_i$ and $a$ with a transformation value $v_i$. The cell values associated with the source member versions are multiplied with the respective transformation value, the multiplication results are summed up. The result is a new cell value associated with the new member version. A merge can, for instance, describe the unification of the three departments $X_1$, $X_2$ and $X_3$ into a single department $X$. The prior departments stop to exist, and $X$ cannot be seen as direct successor (in the sense of legal successor) of any of them. The transformation values could express the fractions which should be summed up into $X$.

There is one problem with the semantics of split and merge as defined above: They are both not preserving the identity of the transformed dimension members. This may be desired in some cases, but in other cases it may not. For instance, if there is a spin-off from a department $X$ which builds an own department $Y$ now, this would be modeled as split of $X$ into $X'$ and $Y$. But of course, $X$ does not stop to exist, but there is just a new version $X'$ of $X$. A similar case can be constructed for the other way round: If some department $Y$ becomes part of department $X$, this would be modeled as merge of $X$ and $Y$ into $X'$. But, of course, $X'$ is not a new entity, but just a new version of $X$. With the split and merge operations as defined above it is not possible to express these desired semantics. So we define two additional operations, which are identity preserving variations of split and merge.

**Spin-off.** The spin-off is an identity-preserving variant of the split. It divides one source member version $a$ from structure version $u$ into a set of destination member versions $\{a_1, \dots, a_n\}$ of structure version $u + 1$. Exactly one of the $a_i \in \{a_1, \dots, a_n\}$ is defined to be a version of the same dimension member as $a$. A spin-off consists of $n$ transformation functions, one between $a$ and each $a_i$ with a transformation value $v_i$.

**Absorption.** The absorption is an identity-preserving variant of the merge. It combines a set of source member versions $\{a_1, \dots, a_n\}$ of structure version $u$ into a single destination member version $a$ of structure version $u + 1$. Exactly one of the $a_i \in \{a_1, \dots, a_n\}$ is defined to be a version of the same dimension member as $a$. A merge consists of $n$ transformation functions, one between each $a_i$ and $a$ with a transformation value $v_i$.

All operations presented so far do transform between two single members or one single member on one side and a set of members on the other side. Many-to-many transformations are still not possible. Thus, as final transformation operation we also define this case:

**Mixed Transformation.** The mixed transformation transforms a set of source member versions $\{a_1, \dots, a_k\}$ of version $u$ into a set of destination members versions $\{a_m, \dots, a_q\}$ of version $u+1$. For each source member $a_i$ there is at most one destination member $a_n$ such that $a_i$ and $a_n$ are versions of the same member. The mixed transformation consists of at most $k \cdot (q - m + 1)$ transformation functions between $a_i \in \{a_1, \dots, a_k\}$ and $a_n \in \{a_m, \dots, a_q\}$.

With this set of transformation operations, we think all possible data transformations can be modelled. The deletion and insertion of a dimension member, which do not imply a change of cell data on the base level do not need to be modelled on this stage.

$$2DC_1 = \begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \end{array} \begin{array}{c} \begin{array}{cc} b_1 & b_2 \end{array} \\ \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix} \end{array}$$

$$4DC_1 = \begin{array}{c} a_1 \\ \\ a_2 \end{array} \begin{bmatrix} \begin{array}{cc} b_1 & b_2 \\ \begin{array}{c} d_1 \; d_2 \\ \begin{array}{c} c_1 \\ c_2 \\ c_3 \end{array} \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \end{array} & \begin{array}{c} d_1 \; d_2 \\ \begin{array}{c} c_1 \\ c_2 \\ c_3 \end{array} \begin{bmatrix} 7 & 8 \\ 9 & 10 \\ 11 & 12 \end{bmatrix} \end{array} \\ \begin{array}{c} d_1 \; d_2 \\ \begin{array}{c} c_1 \\ c_2 \\ c_3 \end{array} \begin{bmatrix} 13 & 14 \\ 15 & 16 \\ 17 & 18 \end{bmatrix} \end{array} & \begin{array}{c} d_1 \; d_2 \\ \begin{array}{c} c_1 \\ c_2 \\ c_3 \end{array} \begin{bmatrix} 19 & 20 \\ 21 & 22 \\ 23 & 24 \end{bmatrix} \end{array} \end{array} \end{bmatrix}$$

**Fig. 1.** Examples for two- and four-dimensional data matrices

### 3.3 Matrix Representation of Data Transformation

For the specification of transformation operations, we first of all, have to define a representation for the cell data and for the transformation operations. In this section we present a matrix based representation, which is convenient for specification purposes. Throughout the paper we use the following naming convention: Dimension member $a_i$ is part of dimension $A$, $b_i$ is part of dimension $B$, etc.

The cell data of an $n$-dimensional data cube can be expressed by an $n$-dimensional *data matrix*. The simplest case is a 2-dimensional matrix whose elements are numbers. A 3-dimensional matrix is a matrix whose elements are vectors, a 4-dimensional matrix is a matrix whose elements are matrices themselves, etc. These matrices can be nested as deep as needed. The data cells are defined by the dimension members naming the respective rows and columns. In Fig. 1 the cell defined by the members $a_1$, $b_2$, $c_3$ and $d_2$ contains the value 12.

Also for the transformation operations a matrix representation is possible. A *transformation matrix* is a two-dimensional matrix in any case. The transformation matrix $T^A_{x,y}$ holds all transformation operations for dimension $A$ between structure version $x$ and $y$. The members naming the rows represent the old member versions, the members naming the columns represent the new member versions. The values in the matrix cell hold the transformation weight from the member naming the respective row to the member naming the respective column. Figure 2 shows examples for transformation matrices containing updates ($b_1$ to $b'_1$), merges ($b_1, b_2, b_3$ into $b_{123}$) and splits ($b_1$ into $b_x, b_y, b_z$). Of course, we do not need to have such a transformation matrix for each transformation

$$T^B_{1,2} = \begin{array}{c} \\ b_1 \end{array} \begin{array}{c} b'_1 \\ \begin{bmatrix} 1.2 \end{bmatrix} \end{array} \qquad T^B_{1,2} = \begin{array}{c} \\ b_1 \\ b_2 \\ b_3 \end{array} \begin{array}{c} b_{123} \\ \begin{bmatrix} 0.8 \\ 1.3 \\ 0.7 \end{bmatrix} \end{array} \qquad T^B_{1,2} = \begin{array}{c} \\ b_1 \end{array} \begin{array}{c} \begin{array}{ccc} b_x & b_y & b_z \end{array} \\ \begin{bmatrix} 0.3 & 0.9 & 0.8 \end{bmatrix} \end{array}$$

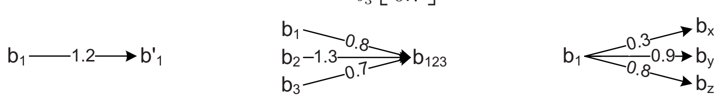



**Fig. 2.** Transformation operations performing member update, merge and split

$$\left[\left(\begin{array}{c|ccc} & b_1 & b_2 & b_3 \\ a_1 & 1 & 2 & 0 \\ a_2 & 0 & 3 & 4 \\ a_3 & 5 & 0 & 6 \end{array} \times \begin{array}{c|cc} & b'_{12} & b'_{23} \\ b_1 & 0.8 & 0 \\ b_2 & 1.2 & 0.9 \\ b_3 & 0 & 1.1 \end{array}\right)^T \times \begin{array}{c|cc} & a'_{12} & a'_3 \\ a_1 & 0.9 & 0 \\ a_2 & 0.7 & 0 \\ a_3 & 0 & 1.3 \end{array}\right]^T \times \begin{array}{c|c} & b'_{123} \\ b'_{12} & 1.1 \\ b'_{23} & 0.9 \end{array} = \begin{array}{c|c} & b'_{123} \\ a'_{12} & 11.871 \\ a'_3 & 13.442 \end{array}$$

(a) All transformation matrices applied from right side

$$\left\{\left[\left(\begin{array}{c|cc} & a'_{12} & a'_3 \\ a_1 & 0.9 & 0 \\ a_2 & 0.7 & 0 \\ a_3 & 0 & 1.3 \end{array}\right)^T \times \begin{array}{c|ccc} & b_1 & b_2 & b_3 \\ a_1 & 1 & 2 & 0 \\ a_2 & 0 & 3 & 4 \\ a_3 & 5 & 0 & 6 \end{array}\right] \times \begin{array}{c|cc} & b'_{12} & b'_{23} \\ b_1 & 0.8 & 0 \\ b_2 & 1.2 & 0.9 \\ b_3 & 0 & 1.1 \end{array}\right\} \times \begin{array}{c|c} & b'_{123} \\ b'_{12} & 1.1 \\ b'_{23} & 0.9 \end{array} = \begin{array}{c|c} & b'_{123} \\ a'_{12} & 11.871 \\ a'_3 & 13.442 \end{array}$$

(b) Transformation matrices applied from left and right

**Fig. 3.** Matrix multiplication for data transformation

operation, but can combine all transformation operations affecting one dimension into a single transformation matrix.

Data transformation using these matrices is done by multiplying the data matrix with the transformation matrices for each dimension by means of standard matrix multiplication. An example is shown in Fig. 3. There are two things to be considered for matrix based data transformation: First of all, one has to be aware of the current transformation matrix's dimension and transpose the data matrix and/or transformation matrix accordingly. In the example, one time the transformation for dimension $A$ is applied from the left and one time from the right side. Thus, the respective matrices have to be transposed accordingly. Second, the order, in which the transformation matrices are applied is irrelevant, as long as the timely dependencies within one dimension are obeyed. Thus, the second transformation matrix for dimension $B$ must not be applied before the first one, but can, of course, be applied before the transformation for dimension $A$. After the transformation matrices for all dimensions have been applied the resulting data matrix is the data matrix containing the data following the new structure version. The standard matrix multiplication does exactly describe the desired transformation for updates, merges, splits and mixed operations.

The transformations presented so far are defined between contiguous structure versions, because transformation operations are only defined between contiguous versions. To transform data between non-contiguous structure versions, we can apply several consecutive transformation matrices for one dimension. In this case the order in which the transformation matrices for one dimension are applied is important: They have to be applied in timely order, thus we must not apply $T^B_{3,4}$ before $T^B_{2,3}$. This timely order does only apply within one dimension, but does not affect application order for different dimensions, i.e. we can apply $T^A_{3,4}$ before $T^B_{2,3}$. In (1) we can see that whether we apply the transformation matrices one after the other on the data matrix (left side), or we calculate a *transitive transformation matrix* by multiplying all matrices first and then apply it on the data matrix (right side) does not matter, as both sides are equal.

$$\{[(C_i \times T^X_{i,i+1}) \times T^X_{i+1,i+2}] \cdots \times T^X_{j-1,j}\} = \{C_i \times [(T^X_{i,i+1} \times T^X_{i+1,i+2}) \cdots \times T^X_{j-1,j}]\} \quad (1)$$

It can be seen from the examples that a transformation matrix has to include *all member versions* of its source structure version in its rows and *all member versions* of its destination structure version in its columns – even the ones which did not change – as otherwise neither the multiplication with the data matrix nor the multiplication of transformation matrices for calculating a transitive transformation matrix can work.

### 3.4 Graph-Based Representation of Data Transformation

The matrix based representation is an adequate method for the specification of the transformation output. But data transformation cannot be implemented efficiently using matrices. For that purpose we defined an alternative, graph based, representation for cell data and transformation operations. The main advantage of the graph based representation is that only cell data which is subject to changes has to be included in the transformation, whereas with the matrix based approach one needs to handle the complete data set, for both, the data matrix and the transformation matrix.

The transformation operations can be modeled by weighted directed acyclic graph (DAG), where the nodes denote the dimension member versions and the edges denote the transformation functions an operation is composed from. The weight of the edges denotes the transformation weight. Examples for transformation graphs can be seen in Fig. 2, where each graph models exactly the same transformation operation as the matrix above it. For transformations over more than one structure version, i.e. a member version resulting from a transformation operation is transformed once again, such transformation graphs can be linked up for one dimension.

The complete cell transformation is also modeled by a DAG, where a node in the graph denotes a single cell value and the edges denote the transformation functions which have to be applied on the cell data. The member transformation graphs are applied on each cell, which is defined by one of the transformed members in the graph. Again, the order of dimension does not matter. The only thing to consider is that the timely order within one dimension is not violated. Each edge in a transformation graph means a multiplication, i.e. the cell value of the cell at the origin of the edge has to be multiplied with the edge weight. Each cell with more than one incoming edge means an addition, as the values from different pathes have to be added. An example for such a transformation graph can be seen in Fig. 4, which does exactly the same transformation as shown in Fig. 3. The nodes on the left side (solid line) of the graph represent the initial cell data, the nodes with the dotted line are the cell values after the first transformation for dimension $B$, the nodes with the dashed line represent the cell values after transformation for dimension $A$, i.e. after the complete transformation from version 1 to 2. The rightmost nodes with the dashes and dots represent the the cell values after the second transformation in dimension $B$. The second transformation, i.e. for dimension $A$, can be done at any other position, e.g. first, or last. The only dependency to obey is that the second transformation for dimension $B$ must not be done before the first one.

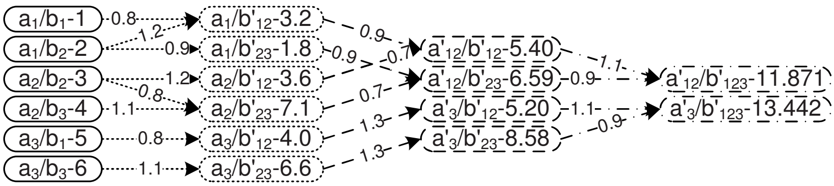



**Fig. 4.** Cell Transformation Graph

As it can be seen from this simple transformation graph, there are transformation operations which reduce the number of data cells (Merge, Absorption), transformation operations which increase the number of data cells (Split, Spin-off) and transformation operations which do not change the number of data cells (Update). The aim is, of course, to have a minimal cell transformation graph, i.e. a cell transformation graph with the least number of nodes and edges. The graphed representation of the data transformation together with the methods from the Algebraic Graph Transformation field [14] offers potential for minimizing the graphs by applying transformation operations in a reasonable order. We also did a prototypic implementation of the graph based transformation and optimization for evaluation purposes, but due to space limitations, we cannot include these results here.

A simple but yet illustrating example could be a company selling bottled water. To keep track of their sales, they have a DWH cube with the three dimensions "Time", "Product" and "Measures". Until 2006 they did not distinguish between mineral water and still water, so the "Product" dimension only contains the member "Water". In the "Measures" dimension the number of "Bottles Sold" is stored in units of 1000 bottles. As of 2007 the company decides to distinguish between "Mineral Water", which is 40% of the sales and "Still Water", which is 60%. Furthermore, the unit is not 1000 bottles any more, but 1.000.000 bottles. So, without handling changes, a report for 2006 and 2007 could look like the left chart in Fig. 5. Such a report cannot be interpreted without knowledge about the recent changes. When applying means of temporal data warehousing, the transformation operation for the "Product" dimension is a *Split* of "Water" into "Mineral Water" and "Still Water". In the "Measures" dimension an *Update* for the member "Bottles Sold" with transformation factor 0.001 has to be done. The resulting report may then look like the right chart in Fig. 5, where the data is easily comparable.

| Bottles Sold | Time | |
|---|---|---|
| Product | 2006 | 2007 |
| Mineral Water | | 4.8 |
| Still Water | | 7.2 |
| Water | 10000 | |

*Update "Bottles Sold"*
$\Longrightarrow$
*Split "Water"*

| Bottles Sold | Time | |
|---|---|---|
| Product | 2006 | 2007 |
| Mineral Water | 4.0 | 4.8 |
| Still Water | 6.0 | 7.2 |

**Fig. 5.** Transformation Example

## 4 Conclusions

We live in a changing world. With changes in reality also the semantics of model elements representing this real world may change, and with them the semantics of the data defined by these models. DWH model aspects of a certain application domain. In this paper we present a way to express changes in the semantics of a DWH structure and how to deal with the impact of these changes on the contained cell data. We present a set of transformation operations and describe two different representations for them. The matrix based representation is intended for the straightforward specification of transformation results only. This representation is not destined for implementation, because the involved matrices (especially the data matrix) become huge. The graph based representation is more efficient – only changed data has to be included – and furthermore offers potential for optimization, i.e. graph minimization. Due to the huge amount of data which has to be dealt with during transformation, such optimization techniques are very important.

## References

1. Statistical Office of the European Communities, http://ec.europa.eu/eurostat/
2. Eder, J., Wiggisser, K.: Data Warehouse Maintenance, Evolution and Versioning. In: Liu, L., Özsu, T. (eds.) Encyclopedia of Database Systems. LNCS. Springer, Heidelberg (2009)
3. Eder, J., Koncilia, C., Morzy, T.: The COMET Metamodel for Temporal Data Warehouses. In: Pidduck, A.B., Mylopoulos, J., Woo, C.C., Ozsu, M.T. (eds.) CAiSE 2002. LNCS, vol. 2348, pp. 83–99. Springer, Heidelberg (2002)
4. Eder, J., Wiggisser, K.: A DAG Comparison Algorithm and its Application to Temporal Data Warehousing. In: ER 2006, pp. 217–226. Springer, Heidelberg (2006)
5. Kimball, R.: Slowly Changing Dimensions. DBMS Magazine 9(4), 14 (1996)
6. Quix, C.: Repository Support for Data Warehouse Evolution. In: Proc. of the $1^{st}$ Int'l. WS on Design and Management of Data Warehouses, p. 4 (1999)
7. Vaisman, A., Mendelzon, A.: A Temporal Query Language for OLAP: Implementation and a Case Study. In: Proc. of the Int'l. WS on Database Programming Languages, pp. 78–96 (2001)
8. Body, M., Miquel, M., Bédard, Y., Tchounikine, A.: Handling Evolutions in Multidimensional Structures. In: Proc. of the $19^{th}$ ICDE, pp. 581–591 (2003)
9. Golfarelli, M., Lechtenbörger, J., Rizzi, S., Vossen, G.: Schema versioning in data warehouses: Enabling cross-version querying via schema augmentation. Data- and Knowledge Engineering 59(2), 435–459 (2006)
10. Malinowski, E., Zimányi, E.: A conceptual solution for representing time in data warehouse dimensions. In: Proc. of the $3^{rd}$ Asia-Pacific Conference on Conceptual Modelling, pp. 45–54 (2006)
11. Kaas, C., Pedersen, T., Rasmussen, B.: Schema Evolution for Stars and Snowflakes. In: Proc. of the $6^{th}$ ICEIS, pp. 425–433 (2004)
12. SAP Inc.: Multi-Dimensional Modeling with BW: ASAP for BW Accelerator. Technical report, SAP Inc. (2000)
13. Kalido: KALIDO Dynamic Information Warehouse: A Technical Overview. Technical report, Kalido (2004)
14. Ehrig, H., Ehrig, K., Prange, U., Taentzer, G.: Fundamentals of Algebraic Graph Transformation. Monographs in Theoret. Comp. Science (2006)

# Managing the History of Metadata in Support for DB Archiving and Schema Evolution

Carlo A. Curino[1], Hyun J. Moon[2], and Carlo Zaniolo[2]

[1] Politecnico di Milano
carlo.curino@polimi.it
[2] University of California, Los Angeles
{hjmoon,zaniolo}@cs.ucla.edu

**Abstract.** Modern information systems, and web information systems in particular, are faced with frequent database schema changes, which generate the necessity to manage them and preserve the schema evolution history. In this paper, we describe the *Panta Rhei Framework* designed to provide powerful tools that: (i) facilitate schema evolution and guide the Database Administrator in planning and evaluating changes, (ii) support automatic rewriting of legacy queries against the current schema version, (iii) enable efficient archiving of the histories of data and metadata, and (iv) support complex temporal queries over such histories. We then introduce the *Historical Metadata Manager (HMM)*, a tool designed to facilitate the process of documenting and querying the schema evolution itself. We use the schema history of the Wikipedia database as a telling example of the many uses and benefits of *HMM*.

## 1 Introduction

The main goal of a traditional database, i.e., capturing the "current state of the modelled reality", is satisfactory only in simple applications [1]. More sophisticated information systems require a systematic archiving and management of the history of the database. In particular, in web information systems, such as Wikipedia, ethical and legal expectations for accountability and preservation arise for information that was once placed in the public domain and divulged to the world; thus policies and systems are needed to support historical archives for web information. This situation has generated renewed interest in broad research on temporal databases and on transaction-time databases, in particular. In addition to data evolution [2], these systems experience intense evolution of database schema, as we reported in our analysis [3] of the Wikipedia DB, which has experienced more than 170 different schema versions in its 4.5 years of lifetime. Schema evolution, already a serious problem in traditional information systems [4,5], becomes even more critical in web information systems.

In order to effectively manage information systems under the realistic assumption that *Panta Rhei—everything is in flux*—so that not only data, but also schemas, queries and applications are in continuous evolution, we propose the *Panta Rhei Framework*[1]. This framework provides a unified solution to the

[1] The project homepage is: http://yellowstone.cs.ucla.edu/schema-evolution/

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 78–88, 2008.

problems of: (i) graceful database evolution in the presence of schema changes, (ii) transaction-time history archiving and querying of databases under schema evolution.

The first objective is achieved by $\mathcal{PRISM}$ [6,7], a tool that assists the Database Administrator (DBA) in designing a new schema and that automatically revises legacy queries to work on it. The second objective is achieved by the transaction-time database system *ArchIS* [8,9], and its extension $\mathcal{PRIMA}$ [10]: *ArchIS* supports efficient temporal queries on the archived history of (fixed-schema) databases, while $\mathcal{PRIMA}$ introduces the transparent support for complex temporal queries over archives with evolving schema. Both $\mathcal{PRISM}$ and $\mathcal{PRIMA}$ exploit an intuitive operational language of Schema Modification Operators (SMOs) to explicitly capture the semantics of the schema evolution.

In this paper, we further extend the framework by introducing the *Historical Metadata Manager (HMM)*, a tool capable of archiving and querying rich metadata histories. Most modern DBMS provide their meta-level information in the form of a virtual DB named `information_schema` (SQL:2003 standard). With the *HMM* we extend the scope of the `information_schema`, inasmuch as the complete history of the DB schema, not just its current snapshot, is preserved and queried. Moreover, the *HMM* exploits the SMO-based representation of the schema evolution to provide users with a better understanding of the semantics of each schema change.

**Contributions.** The novel contributions of this paper are the following:

- the overall architecture of the *Panta Rhei Framework* for graceful schema evolution and historical archive management under schema evolution;
- the *History Metadata Manager*, a tool capable of providing an effective archival mechanism for metadata and a query facility enabling complex temporal queries on schema histories.

The paper is organized as follows: Section 2 provides an overview of the *Panta Rhei Framework* architecture. Section 3 describes the *Historical Metadata Manager (HMM)* and its capabilities. In Section 4, we discuss related works, while Section 5 is devoted to future developments. Section 6 draws our conclusions.

## 2 The *Panta Rhei Framework*

The *Panta Rhei Framework*, presented in Figure 1, aims at providing seamless support for evolution of both data and schema and complete history management. The objectives of the framework are as follows: (i) a workbench to assist the DBA designing schema revisions with tools for change impact analysis and automatic query rewriting, (ii) efficient archiving of DB histories, (iii) support for complex temporal queries over a historical archive under schema evolution, and (iv) archiving and temporal querying of the metadata history.

Objective (i—iii) were largely realized by $\mathcal{PRISM}$ [6,7], *ArchIS* [8,9] and $\mathcal{PRIMA}$ [10,11], but this paper is the first to describe the overall architecture of our *Panta Rhei Framework* and the design of the *History Metadata Manager (HMM)* addressing objective (iv).

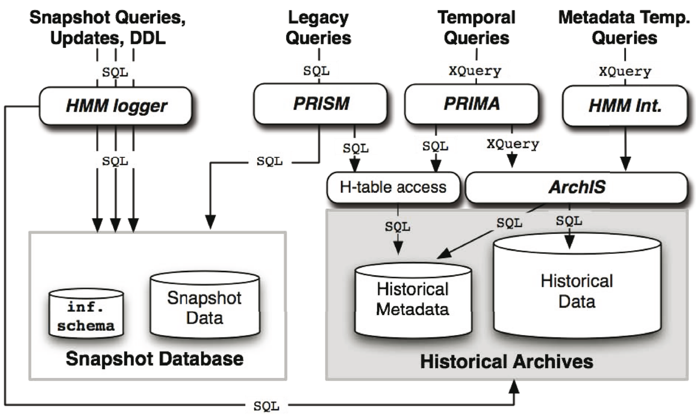



**Fig. 1.** *Panta Rhei Framework* Architecture

### 2.1 *PRISM*

*PRISM* aims at automating the error-prone and time-consuming activity of schema evolution. The system provides: (i) a language of Schema Modification Operators (SMO) used for expressing complex schema changes in a concise way, (ii) tools that allow the DBA to evaluate the effects of schema changes, (iii) automated data migration, and (iv) optimized support for legacy queries on the current schema version.

SMOs provide an operational and concise way to capture schema evolution [3,6], that specify how both schema and data evolve. By analyzing SMO sequences specified by the DBA, the system provides feedback on the proposed evolution, thus improving predictability of the evolution process. The same SMO representation is used to automate data migration and query adaptation by (i) generating SQL data migration scripts, and (ii) automatically rewriting legacy queries to operate on the new schema. Legacy queries are automatically supported in two alternative ways: by means of SQL views (among schema versions) or by automatically translating queries across schema versions by the query rewriting engine [12]. Wikipedia and its 170+ schema versions provided an invaluable testbed for validating *PRISM* and its query rewriting capabilities.

### 2.2 ArchIS

*ArchIS* [9] provides, in the context of *Panta Rhei Framework*, a powerful temporal data model based on XML, called V-Documents. V-Documents represent, in a temporally-grouped fashion, the attribute-level-timestamped history of the snapshot DB. This simplifies issues such as temporal coalescing [13]; moreover, the choice of XML allows the use of XQuery, which proved well-suited to express complex temporal queries [8]. To overcome the current performance limitations of XML, *ArchIS* exploits mature RDBMS technology, by shredding V-Documents and storing them in a relational format named *H-tables*. It also translates XQuery into the equivalent SQL/XML queries that can be executed over *H-tables*.

### 2.3 *PRIMA*

*PRIMA* [10,11] extends the functionalities of *ArchIS* by allowing the schema to evolve as data evolve. At every schema change, the snapshot data are archived using the schema version under which they first appeared, to achieve perfect archival quality[2]. This produces a transaction-time database that archives the data history under multiple schema versions. On the other hand, the task of writing temporal queries on such archive might become taxing for the users, since queries may potentially span over several schema versions (tens or even hundreds in the Wikipedia scenario). *PRIMA* addresses this issue by letting the users express temporal queries under a selected schema version, typically the current one, and then automatically rewriting them against pertinent schema versions.

### 2.4 Putting All Together: The Overall Architecture

The overall architecture of the *Panta Rhei Framework* is shown in Figure 1. The left-hand side of Figure 1 represents a snapshot database, augmented by the logging functionality, which updates the historical archives of both data and metadata (Section 3.3). The right-hand side of Figure 1 shows the core components of our architecture: *PRISM*, *ArchIS*, *PRIMA*, and the *HMM* querying interface. These components share a common data layer based on two archives, capturing data and metadata histories, and the SMO representation of the schema evolution. In particular, the *Historical Data DB* represents the transaction-time archive of the snapshot database—maintained by *ArchIS* in the relational format of *H-table*. In a similar way, the *Historical Metadata DB (HMDB)* stores the history of the meta-information that the SQL-compliant DBMSs offer in the `information_schema` and the SMO-based description of the schema changes. These data, besides being used by *PRISM* and *PRIMA*, provide invaluable documentation when queried by the *Historical Metadata Manager (HMM)*, as further discuss in Section 3. One of the benefits of this integrated solution is that the effort needed to adopt one component, e.g., *PRISM*, enables at no extra cost the entire set of functionalities of the *Panta Rhei Framework*.

## 3 Historical Metadata Manager

The *Historical Metadata Manager (HMM)* is a tool capable of storing and querying metadata histories. In this section, we discuss its main components, namely (i) the underlying data layer *Historical Metadata DB (HMDB)*, (ii) the querying

[2] Data migration is, in general, not information-preserving, thus can potentially compromise archival quality.

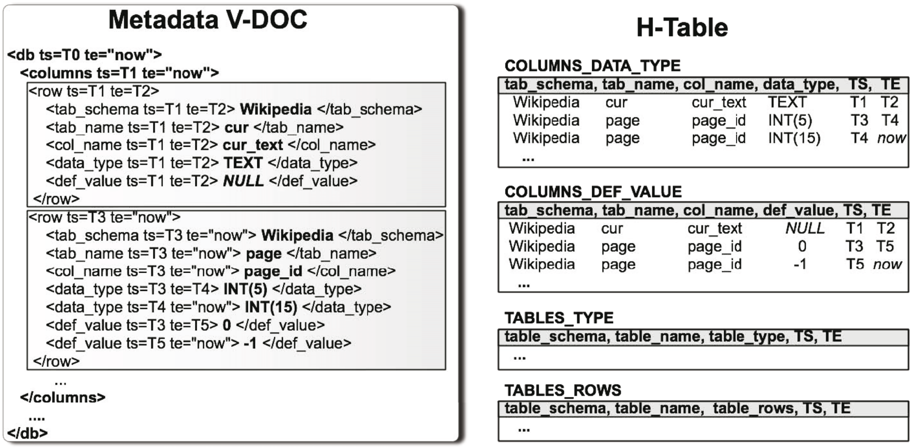



**Fig. 2.** *Historical Metadata DB*: a portion of V-Document and the corresponding H-table representations

interface allowing to pose complex temporal queries over the *HMDB*, and (iii) the *HMM logger*, a component that maintains the historical archives updated, based on the set of changes occurred in the snapshot data and metadata.

### 3.1 Historical Metadata DB

The *HMDB* captures the history of the `information_schema`, together with additional metadata needed in the *Panta Rhei Framework* (SMOs, user queries, etc.). Due to space limitation, we discuss only a selected subset of the *HMDB* content. We introduce it in a snapshot format, to later present the corresponding temporal archive.

```
schemata(schema_name, default_character_set_name)
tables(table_schema, table_name, table_type, table_rows)
columns(table_schema, table_name, column_name, data_type, default_value)
queries(query_id, query_text, schema_name, issue_timestamp)
query_exec(query_id, exec_timestamp, exec_success)
smos(smo_id, smo_text, smo_type, timestamp)
```

The first three relations, `schemata`, `tables`, and `columns`, come from the `information_schema`, while the relations named `queries`, `query_exec`, and `smos` store additional metadata needed by $\mathcal{PRISM}$ and $\mathcal{PRIMA}$. The `queries` relation archives the user queries (as templates), the target schema, and query issuing time. The table `query_exec` stores the results (boolean flag `exec_success`) of testing each of above query templates against the subsequent schema versions, used in the $\mathcal{PRISM}$ impact analysis. Lastly, the table `smos` stores the SMOs describing the semantics of each schema change.

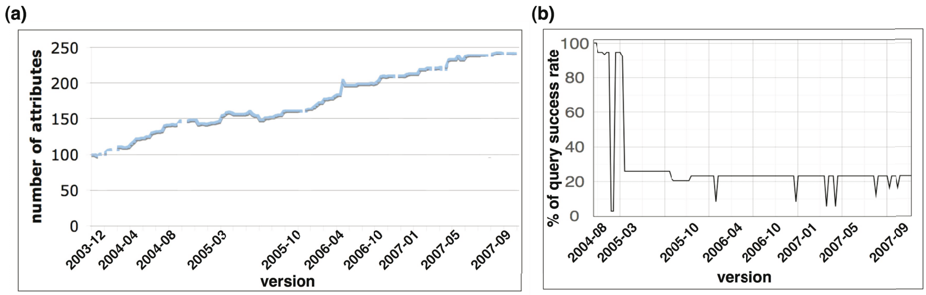



**Fig. 3.** Wikipedia results for: (a) Result of Query 2: Wikipedia schema size, and (b) Result of Query 5: success rate of Wikipedia legacy queries against the subsequent schema version

In order to effectively and efficiently archive the history of these metadata, we represent them in the V-Document format. Figure 2 shows a portion of the V-document, capturing the evolution of two columns in the Wikipedia schema history, and its corresponding *H-table* representation—`ts` and `te` represent respectively the start and end time of the period in which a given value was valid in the snapshot DB. The first row element represents a column `cur.cur_text` which did not change between time T1 and T2, when it disappears forever. The second row element, in contrast, shows a column `page.page_id`, which was newly introduced at time T3, changed its type at time T4 (from INT(5) to INT(15)) and its default value at time T5 (from 0 to -1). The *H-table* representation in Figure 2 highlights how the XML data have been shredded in a set of tables with a regular structure: key-columns, nonkey-column, ts, te.

### 3.2 Querying the *HMDB*

The *HMM Interface* component of Figure 1 builds on top of the *ArchIS* storage engine to provide a simple query interface over the *HMDB*. By means of simple query examples on the Wikipedia schema evolution history, we illustrate how the *HMM* can be leveraged to obtain a deep insight on the evolution itself. The queries we present are grouped in two classes: (i) queries over the history of the `information_schema` (*Query 1-Query 3*), and (ii) queries exploiting SMOs and logged queries (*Query 4* and *Query 5*).

Simple but very useful temporal queries are snapshot queries as the following:

*Query 1. What was the Wikipedia database schema valid on 2007-01-01?*

```
for $db in document("wiki-hmdb.xml")/db,
  $tn in $db/Tables/row/table_name[@ts<="2007-01-01" and @te>"2007-01-01"]
```

Similarly, it is possible (by range queries) to inspect specific portions of the history, retrieving all the subsequent modifications occured in a given period. Next, we give an example of temporal aggregate queries that can be exploited

```
return <table> {$tn/@text}, <cols>
        {$db/Columns/row[table_name=$tn]/
        column_name[@ts<="2007-01-01" and @te>"2007-01-01"]/text()}</cols>
       </table>
```

to observe general trends of the schema evolution:

*Query 2. Retrieve the number of columns in the Wikipedia DB throughout the history.*

```
for $db in document("wiki-hmdb.xml")/db,
     $t in $db/Schemata/rows/timestamp
return <tuple> <timestamp>{$t}</timestamp>,
        <cnt>{count($db/Columns/row/column_name[@ts<=$t and @te>$t])}</cnt>
       </tuple>
```

The output of such query is nicely rendered in Figure 3a, where it is easy to spot a net growing trend in the Wikipedia schema size, roughly 31% a year.

Furthermore, to analyze quality and stability of the design it is possible to pose queries retrieving stable or unstable portions of the schema as the following:

*Query 3. Which table in Wikipedia DB schema that remained valid for the longest period?*

```
let maxInterval:=
      max(for $tn in document("wiki-hmdb.xml")/db/Tables/row/table_name
          return $tn/@te-$tn/@ts)
for $tn in document("wiki-hmdb.xml")/db/Tables/row/table_name
where $tn/@te-$tn/@ts = maxInterval
return $tn/text()
```

The answer of this query reveals how the *user* table was the most stable in the DB schema. Another interesting class of queries exploits the explicit characterization of the change semantics provided by SMOs to track back (or forward) the evolution of information stored in a given schema element. Consider the following example:

*Query 4. Retrieve the previous version of the information currently stored in table 'page'.*

```
let $last_smo_timestamp := max(document("wiki-hmdb.xml")/db/Tables/row/
      smo[affected_table_name="page"]/timestamp)
let $last_smo := document("wiki-hmdb.xml")/db/Tables/row/
      smo[affected_table_name="page" and timestamp=$last_smo_timestamp]
return $last_smo/input_tables
```

Finally, by exploiting the information about query templates and their execution, it is possible to retrieve and visualize the impact of the schema evolution on the original user queries, as exemplified by the following query:

*Query 5. What's the success rate of legacy queries (valid on 2004-08-15) after each Wikipedia schema change?*

```
for $db in document("wiki-hmdb.xml")/db,
    $t in $db/Schemata/rows/timestamp
return
 <tuple>
   <schema-change-time>{$t}</schema-change-time>,
   <success-rate>{avg($db/query_exec/row
[query_id=$db/queries/row[execution_timestamp="2004-08-15"]/query_id]/
           success_flag)}</success-rate>
 </tuple>
```

The result of the query is shown in Figure 3b. The graph effectively highlights that a sudden drop occurred around 2005-03, due to a deep change of the article revision management in the Wikipedia DB [3]; this impacted over 70% of the queries. The spikes in the graph corresponds to syntactically incorrect schema versions, for which most of the queries failed, see [3] for details.

These relatively simple queries are naturally supported by the *HMM* on the history of metadata we archive. This querying capability provides a powerful tool for dissecting the metadata history, enabling a better understanding of undergoing schema evolution.

### 3.3 Archive Maintenance

This section briefly discusses how we maintain the historical archives up to date. The *HMM logger* of Figure 1 integrates the following functionalities:

**Data Archive Update.** The problem of creating and maintaining a temporal archive of the snapshot DB has been addressed in [9]. Two solutions have been proposed, one based on active rules and the other on update logs. In the *ArchIS-DB2* experience, a set of DML triggers have been exploited to maintain the *H-table* archive updated by propagating every update of the snapshot DB to the temporal archive, while the *ArchIS-ATLaS* prototype obtains the same results by processing the update log. The choice between the two approaches depends on performance issues, the interested reader can refer to [9].

**Metadata Archive Update.** To adapt the above methods to work on metadata, we exploit DDL triggers and SQL DDL logs analysis. The `information_schema` is, in fact, a virtual DB and no direct updates are issued on it. Recent versions of Oracle and MS SQL Server support DDL triggers. For MySQL and DB2, simple workarounds have been designed, based on the fact that the `information_schema` provides timestamps of the last modifications and that the frequency of schema changes is rather low. It is thus possible to monitor[3] it to detect changes, and use the timestamps stored in the snapshot `information_schema` itself, to obtain an accurate temporal archive. The *HMDB* also stores the sequences of Schema Modification Operators describing the semantics of the schema change. This is achieved in two alternative ways: (i) by systematically

[3] This can be automated within the DBMS by means of *event* triggers (MySQL 5.1) or scheduled administration tasks (DB2).

exploiting $\mathcal{PRISM}$ to explicitly model the schema evolution of the snapshot DB, or (ii) by observing the evolution and semi-automatically mining the corresponding SMOs.

## 4 Related Works

The long standing problems of schema evolution and versioning [14,15] have been the subject of several research efforts surveyed in [16], and more recent approaches were proposed in [17,18,19,20]. Archiving and querying metadata histories has been discussed, to a limited extent, in [21], as a byproduct of temporal data archiving. Recent theoretical and practical advances on the related problems of query rewriting, mapping composition and invertibility appeared in [12,22,23,24]. The *Panta Rhei Framework* builds on top of this solid theoretical foundations and provides, at the best of our knowledge, one of the most advanced, unified approaches to the issues of schema evolution and data and metadata history management.

## 5 Future Developments

Much of the *Panta Rhei Framework* is still under development; along with the implementation of new features and optimizations techniques, a substantial integration effort is taking place to provide an easily-deployable, unified tool-suite.

Mapping and query rewriting are two core features that our *Panta Rhei Framework* shares with most of the existing approaches to the problem of data integration. It is part of our research agenda to investigate the use of the techniques we exploit in $\mathcal{PRISM}$ and $\mathcal{PRIMA}$ to tackle data integration and schema evolution in a unified way, as we discuss in [25]. Moreover, metadata histories can be effectively exploited in the context of automatic schema matching to develop history-aware matching heuristics. Further investigation is on-going about the use of semantic data models, to capture uniformly schema changes and schema mappings [26] in a semantically rich model, which might enable automatic reasoning.

## 6 Conclusions

In this paper, we presented the overall architecture of the *Panta Rhei Framework*, that unifies the concurrently ongoing endeavors on temporal databases and schema evolution that have produced $\mathcal{PRISM}$ [6,7], *ArchIS* [8,9], and $\mathcal{PRIMA}$ [10,11]. Thus, this is the first paper describing the *History Metadata Manager (HMM)*, a powerful tool we have developed for archiving and querying the history of the metadata evolution. Such information, representing the core knowledge on which $\mathcal{PRISM}$ and $\mathcal{PRIMA}$ operate, is invaluable for a Database Administrator trying to understand the schema evolution of an Information System. The unified XML model exploited allows the use of XQuery, which is well-suited to express complex temporal queries. The *Panta Rhei Framework* has been validated on the real-life schema evolution of the Wikipedia database.

## References


1. Snodgrass, R.T.: The TSQL2 Temporal Query Language. Kluwer, Dordrecht (1995)
2. Almeida, R.B., Mozafari, B., Cho, J.: On the evolution of wikipedia. In: Int. Conf. on Weblogs and Social Media (March 2007)
3. Curino, C.A., Moon, H.J., Tanca, L., Zaniolo, C.: Schema Evolution in Wikipedia: toward a Web Information System Benchmark. In: ICEIS (2008)
4. Sjoberg, D.I.: Quantifying schema evolution. Information and Software Technology 35(1), 35–44 (1993)
5. Marche, S.: Measuring the stability of data models. European Journal of Information Systems 2(1), 37–47 (1993)
6. Moon, H.J., Curino, C.A., Deutsch, A., Hou, C.Y., Zaniolo, C.: Managing and querying transaction-time databases under schema evolution. In: VLDB (2008)
7. Curino, C.A., Moon, H.J., Ham, M., Zaniolo, C.: Architecture and optimization for transaction-time dbs with evolving schemas. Demo proposal submitted for publication (2009)
8. Wang, F., Zaniolo, C.: An XML-Based Approach to Publishing and Querying the History of Databases. World Wide Web: Web Information Systems Engineering 8(3), 233–259 (2005)
9. Wang, F., Zaniolo, C., Zhou, X.: Archis: An xml-based approach to transaction-time temporal database systems. The International Journal of Very Large Databases (2008)
10. Curino, C.A., Moon, H.J., Zaniolo, C.: Graceful database schema evolution: the prism workbench. In: VLDB (2008)
11. Moon, H.J., Curino, C.A., Zaniolo, C.: Architecture and optimization for transaction-time dbs with evolving schemas (submitted for publication, 2009)
12. Deutsch, A., Tannen, V.: Mars: A system for publishing XML from mixed and redundant storage. In: VLDB (2003)
13. Clifford, J., Croker, A., Grandi, F., Tuzhilin, A.: On Temporal Grouping. In: Recent Advances in Temporal Databases, pp. 194–213. Springer, Heidelberg (1995)
14. Roddick, J.: A Survey of Schema Versioning Issues for Database Systems. Information and Software Technology 37(7), 383–393 (1995)
15. Castro, C.D., Grandi, F., Scalas, M.R.: Schema versioning for multitemporal relational databases. Information Systems 22(5), 249–290 (1997)
16. Ram, S., Shankaranarayanan, G.: Research issues in database schema evolution: the road not taken. Boston University School of Management, Paper No: 2003-15 (2003)
17. Dyreson, C., Snodgrass, R.T., Currim, F., Currim, S., Joshi, S.: Validating quicksand: Schema versioning in auxschema. icdew 0, 82 (2006)
18. Rizzi, S., Golfarelli, M.: X-time: Schema versioning and cross-version querying in data warehouses. In: ICDE, pp. 1471–1472 (2007)
19. Grandi, F., Mandreoli, F., Tiberio, P.: Temporal modelling and management of normative documents in xml format. Data Knowl. Eng. 54(3), 327–354 (2005)
20. Jørgensen, P.S., Böhlen, M.H.: Versioned relations: Support for conditional schema changes and schema versioning. In: Kotagiri, R., Radha Krishna, P., Mohania, M., Nantajeewarawat, E. (eds.) DASFAA 2007. LNCS, vol. 4443, pp. 1058–1061. Springer, Heidelberg (2007)
21. Wang, F., Zaniolo, C.: Representing and Querying the Evolution of Databases and their Schemas in XML. In: Intl. Workshop on Web Engineering, SEKE (2003)

22. Nash, A., Bernstein, P.A., Melnik, S.: Composition of mappings given by embedded dependencies. In: PODS (2005)
23. Bernstein, P.A., Green, T.J., Melnik, S., Nash, A.: Implementing mapping composition. VLDB J. 17(2), 333–353 (2008)
24. Fagin, R.: Inverting schema mappings. ACM Trans. Database Syst. 32(4) (2007)
25. Curino, C.A., Tanca, L., Zaniolo, C.: Information systems integration and evolution: Ontologies at rescue. In: International Workshop on Semantic Technologies in System Maintenance (STSM) (2008)
26. Artale, A., Parent, C., Spaccapietra, S.: Evolving objects in temporal information systems. Ann. Math. Artif. Intell. 50(1-2), 5–38 (2007)

# Towards a Dynamic Inconsistency-Tolerant Schema Maintenance

Hendrik Decker*

Instituto Tecnológico de Informática, Valencia, Spain
hendrik@iti.es

**Abstract.** When a relational database schema changes, the questions arises if any integrity constraint is violated by the change. For dynamic schema maintenance, traditional methods to answer this question have several debilities. First, they require that all integrity constraints be satisfied before admitting any update, although extant integrity violations are frequent in practice. Second, they are inefficient for dynamic changes of integrity constraints. Third, they are unflexible wrt safety-critical constraints. Fourth, they usually do not care at all whether an updated schema remains satisfiable. We propose improvements of each of these weaknesses.

## 1 Introduction

The field of database schema evolution has attracted an unrelented growth of interest in recent years [12]. However, schema updates that include changed integrity constraints have received only scant attention. A notable exception is [3]; cf. also [15], where the "semantic correctness" of evolving schemas is mentioned as an open issue. This lack of attention is surprising, since most database schemas involve a set of constraints, for delimiting the space of admissible database states, and this set may evolve over time just like any other part of the schema.

Throughout, we speak of *updates* or *changes* of the schema, rather than of its *evolution*. This is to avoid confusion with other topics traditionally related to schema evolution, such as change propagation, matching and mapping of schemas, versioning or temporal databases. These are not dealt with here.

Inconsistency tolerance is another issue that is enjoying an increase of coverage in the literature [1,16]. A database is inconsistent if some of its integrity constraints are violated. Thus, inconsistency tolerance is the ability to soundly perform database operations also in the presence of violated integrity. Such situations are particularly frequent when schemas evolve dynamically.

In this paper, we look into a part of the underexposed intersection of dynamic schema maintenance and inconsistency-tolerant integrity maintenance. Dynamic schema maintenance deals with simultaneous changes of the database schema, including integrity constraints, and its current state. Inconsistency-tolerant integrity maintenance involves integrity checking [10] and satisfiability checking [2] upon updates of the schema or the state, which yields sound output even in the presence of violated constraints [5,6].

* Partially supported by FEDER and the Spanish MEC grant TIN2006-14738-C02-01.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 89–98, 2008.

In section 2, we ask the reader to consent on some preliminary conventions and definitions. In section 3, we review inconsistency-tolerant integrity checking and extend its definition to arbitrary schema updates. In section 4, we show how integrity can be efficiently maintained for dynamic schema updates while tolerating inconsistencies, even in safety-critical applications and even for unsatisfiable schemas. In section 5, we conclude.

## 2 Preliminaries

Unless specified otherwise, the terminology and notations used are conventional in the logic databases community [13].

### 2.1 Integrity Constraints, Database Schemas, Databases

An *integrity constraint* (in short, *constraint*) is a first-order predicate logic sentence, represented either as a denial (i.e., a clause without head whose body is a conjunction of literals) or in *prenex normal form* (i.e., quantifiers outermost, negations innermost). An *integrity theory* is a set of constraints. Such a theory usually serves to restrict the space of admissible database states, in that each of its constraint is required to be satisfied in each state.

A *database schema* (in short, *schema*) is a triple $S = (T, R, IC)$ where $T$ is a set of *table definitions*, $R$ is a set of *view definitions* and $IC$ is an integrity theory. Each table defined in $T$ uniquely corresponds to a *base predicate* with some arity. Each view defined in $R$ corresponds to a *view predicate*, defined by a set of *rules* that recur on base predicates, as in deductive databases. A *fact* is of the form $p(t_1, ..., t_n)$, where $p$ is either a base predicate or a view predicate, $n$ is the arity of $p$ and each $t_i$ is a ground term. A *database* is a pair $D = (S, E)$ where $S$ is a schema and $E$ is a *state* of $S$, i.e., a set of base predicate facts. Sometimes, $E$ is also called an *extension* of the table definitions of $S$. As usual, we assume that, for each schema $S = (T, R, IC)$ and each database $D = (S, E)$, $E \cup R$ determines a unique minimal Herbrand model, sometimes called the *standard model* of $D$.

The informal definition of 'state' above deliberately does not require anything wrt $IC$, apart from the syntax of constraints. Traditionally, all constraints in $IC$ are required to be *satisfied*, i.e., *true* in the standard model of each state. However, since we are interested in change management also in the presence of inconsistency, the definition above is as general as to allow also for states in which integrity is *violated*, i.e., at least one of the constraints in $IC$ is *false* in $D$.

For a database $D$, an integrity theory $IC$ and an integrity constraint $I$, let $D(IC) = sat$ and $D(I) = sat$ denote that $IC$ or, resp., $I$ is satisfied in $D$, and $D(IC) = vio$ ($D(I) = vio$) that it is violated. 'Consistency' and 'inconsistency' are synonymous with 'satisfied' and, resp., 'violated' integrity.

### 2.2 Updates

An *update* $U$ is a function that maps databases to databases. For an update $U$, a schema $S = (T, R, IC)$ and a database $D = (S, E)$, let $D^U = (S^U, E^U)$ denote the database, i.e., the schema and the state, into which $D$ is mapped by $U$; let $T^U$, $R^U$, $IC^U$ be the definitions of tables, views and, resp., integrity constraints in $S^U$. If $D = (S, E)$ is updated by $U$, then $D$ (resp., $S$, $E$) is also called the *old* database (resp., schema, state), and $D^U$ (resp., $S^U$, $E^U$) the *new* database (resp., schema, state). If $IC = IC^U$, then $U$ is called a *conservative update*. For an arbitrary update $U$, let $U_c$ denote the maximal subset of $U$ that is a conservative update.

Equivalently, an update $U$ can be characterized by the symmetric difference of $D$ and $D^U$, i.e., by the tables structures, view definitions, constraints and base predicate facts that either are in the respective components of $D$ but not in those of $D^U$ or vice-versa. Hence, each modification of any component of $D$ can be expressed by the deletion of its old version and the insertion of its new one. However, a representation of modifications as pairs of deletions and insertions is independent of the way modifications are treated in methods that check updates for integrity preservation. In fact, each such method is free to deal with modifications as either one or several operations within a transaction.

The preceding definition of updates covers arbitrary transactions containing simultaneous updates of the schema and the state of a database. Conservative updates include schema changes that leave the integrity theory untouched.

### 2.3 Integrity Checking Methods

From now on, let $\mathcal{M}$, $D$, $U$ and $IC$ always denote an integrity checking method, a database, an update and, resp., an integrity theory. And, if nothing else is said, then $IC$ be always the integrity theory in the schema of $D$.

Each $\mathcal{M}$ can be formalized as a mapping that takes as input a pair $(D, U)$ and outputs upon termination either *ok* or *ko*, where *ok* means that $\mathcal{M}$ sanctions the update and *ko* that it does not. The computation of this output usually involves less access to $E$, and thus is more efficient, than a *brute-force* check, i.e., a plain evaluation of all constraints against the entire database.

For each $\mathcal{M}$, we only consider classes of pairs $(D, U)$ such that the computation of $\mathcal{M}(D, U)$ terminates. In practice, that can always be achieved by a timeout mechanism with output *ko*.

Note that, instead of *ok* and *ko*, we do not use the answers *yes* and *no*, nor the truth values *true* and *false*, nor the integrity values *sat* and *vio*, to denote the output of $\mathcal{M}$. After all, *yes*, *true* or *sat* in place of *ok* could suggest that $D^U(IC^U) = sat$, although $\mathcal{M}$ may tolerate inconsistency and hence output *ok* even if $D^U(IC^U) = vio$. Moreover, if $\mathcal{M}$ is incomplete, then *ko* means that further checks are necessary to determine if integrity is really violated by the update, while *no*, *false* or *vio* would suggest that integrity is indeed violated.

Soundness and completeness of an integrity checking method $\mathcal{M}$ is defined as follows. Note that also changes of the schema of $D$ are taken into account.

**Definition 1.** (*sound and complete integrity checking*)
$\mathcal{M}$ is called *sound* or, resp., *complete* if, for each $(D, U)$ such that $D(IC)$=*sat*, (1) or, resp., (2) holds.

$$\text{If} \quad \mathcal{M}(D, U) = ok \quad \text{then} \quad D^U(IC^U) = sat. \tag{1}$$

$$\text{If} \quad D^U(IC^U) = sat \quad \text{then} \quad \mathcal{M}(D, U) = ok. \tag{2}$$

Although (1) and (2) only relate the output *ok* of $\mathcal{M}$ to integrity satisfaction, symmetric conditions for relating *ko* to integrity violation could be defined. We omit that, since, as is easily seen, soundness and, resp., completeness for *ko* and violation is equivalent to (2) and, resp., (1).

Several methods (e.g., in [2,4,9,11,14]) have been shown to be sound and complete for significant classes of pairs $(D, U)$. Other methods (e.g., in [7,8]) are only shown to be sound. Thus, they provide sufficient but not necessary conditions for guaranteeing integrity of $D^U$.

## 3 Inconsistency-Tolerant Integrity Checking

We are going to we review inconsistency-tolerant integrity checking [5,6] and extend it to cover arbitrary schema updates.

### 3.1 Motivation

In all methods for efficient integrity checking, only certain instances of constraints, called 'relevant cases', are checked. Informally, relevant cases are those that are potentially violated by a given update. By focusing on relevant cases, the cost of integrity checking can be reduced to feasible proportions [10].

This principle can be successfully applied also if some constraints are violated before the update. For example, in a database containing marriage records, suitable constraints can prevent bigamy, married minors, and other unlawful cases. Despite the constraints, however, such a database may, for instance, contain entries of persons married twice, underaged spouses, etc. Such integrity violations may be due to omissions (e.g., divorce from the previous spouse was not recorded), neglect (e.g., the DBA had switched off integrity checking for schema evolution or when migrating to a new system), or other irregularities (e.g., legislation changed after marriage). Yet, new marriages that satisfy all constraints can be entered without problems into the database, even in the presence of inconsistencies. This works well, because only data that are relevant to updates are checked for consistency.

Indeed, it has been shown in [5,6] that several integrity checking methods in the literature function soundly also if integrity is violated before the update. This came as a surprise because the correctness proofs of such methods all rely on the premise that integrity be satisfied before each update. For convenience, let us call this requirement the *total integrity premise*. Methods that continue to work well when this premise is waived are called *inconsistency-tolerant*. Next, we define inconsistency tolerance more formally.

### 3.2 Definition and First Examples

Each constraint $I$ can be conceived as a set of particular instances, called 'cases', of $I$, such that $I$ is satisfied if and only if all of its cases are satisfied. Thus, integrity maintenance can focus on satisfied cases, and check if their satisfation is preserved across updates. Violated cases can be ignored or dealt with at some more convenient moment. This is captured by the following definition, which, as opposed to similar definitions in [5,6], also takes changes of $IC$ into account.

**Definition 2.** (*inconsistency-tolerant integrity checking*)

***a***) A variable $x$ is called *global* in $I$ if $x$ is $\forall$-quantified in $I$ and $\exists$ does not occur left of the quantifier of $x$.

***b***) Let $I$ be of the form $\leftarrow B$ or $QW$, where $B$ is a conjunction of literals, $Q$ is a vector of $\forall$-quantifiers of all global variables in $I$ and $W$ is a formula in prenex normal form, and $\zeta$ be a substitution of the global variables in $I$. Then, $\leftarrow B\zeta$ or, resp., $\forall(W\zeta)$ is called a *case* of $I$, where $\forall(\,.\,)$ denotes universal closure.

***c***) Let $\mathsf{SatCas}(D, IC) = \{\, C : \text{there is } I \in IC,\ C \text{ is case of } I,\ D(C) = sat \,\}$ be the set of cases of constraints in $IC$ that are satisfied in $D$.

***d***) $\mathcal{M}$ is called *sound,* resp., *complete wrt inconsistency tolerance for* $(D, U)$ if, for each $C \in \mathsf{SatCas}(D, IC^U)$, (3) or, resp., (4) holds.

$$\text{If } \mathcal{M}(D,U) = ok \text{ then } D^U(C) = sat. \tag{3}$$

$$\text{If } D^U(C) = sat \text{ then } \mathcal{M}(D,U) = ok. \tag{4}$$

$\mathcal{M}$ is called *sound,* resp., *complete wrt inconsistency tolerance* if $\mathcal{M}$ is sound, resp., complete wrt inconsistency tolerance for each pair $(D, U)$.

We may speak of an 'inconsistency-tolerant method' when it is clear that we actually mean a method that is sound wrt inconsistency tolerance.

Several methods, e.g., those described in [2,9,11,14], are shown to be inconsistency-tolerant for conservative updates in [5,6]. The methods in [2,11] also are complete wrt inconsistency tolerance, as shown in yet unpublished work.

Definition 2 is illustrated by examples 1, 2 and, later, examples 3–6. Examples 1 and 5 feature conservative updates of base facts; 2–4 and 6 are schema updates involving changes of rules or integrity constraints. Example 5 also features the method described in [7], which is not inconsistency-tolerant.

*Example 1.* For tables $p$ and $q$, $I = \leftarrow p(x,y), p(x,z), y \neq z$ is a primary key constraint on the first column of $p$. It is referenced by the foreign key constraint $I' = \forall x,y\, \exists z (q(x,y) \rightarrow p(y,z))$ on the second column of $q$. The global variables of $I'$ are $x$ and $y$; all variables of $I$ are global. For $U = insert\ q(a,b)$, a typical inconsistency-tolerant method $\mathcal{M}$ only evaluates the simplified case $\exists z\ p(b,z)$ of $I'$. If, for instance, $p(b,b)$ and $p(b,c)$ hold in the old state $D$, $\mathcal{M}$ outputs $ok$, ignoring cases that are violated in $D$, such as, e.g., $\leftarrow p(b,b), p(b,c), b \neq c$ and $I$, i.e., extant violations of the primary key constraint. If no fact matching $q(x,b)$ nor any fact matching $p(b,z)$ holds in $D$, then the case $C = \exists z (q(x,b) \rightarrow p(b,z))$ of $I$ is in $\mathsf{SatCas}(D, IC)$ and $\mathcal{M}$ outputs $ko$, indicating that $U$ violates $C$.

*Example 2.* Let $q$ be a ternary base predicate and $p$ be the view which projects $q$ on its first two columns, i.e., $p$ is defined by the rule $p(x,y) \leftarrow q(x,y,z)$. Let $I = \leftarrow p(x,x)$ constrain $p$ to be void of symmetric tuples. Let $q(a,a,b)$, $q(b,b,a)$, $q(b,b,b)$ be the only facts in $D$ that match $q(x,x,z)$. Clearly, the cases $\leftarrow p(a,a)$

and $\leftarrow p(b,b)$ are violated in $D$. Hence, traditional integrity checking refuses to handle the update $U$ which modifies the definition of $p$ to $p(x,z) \leftarrow q(x,y,z)$. As opposed to that, each complete inconsistency-tolerant method $\mathcal{M}$ outputs *ok* if no fact matching $q(x,y,x)$ such that $x \neq a$ and $x \neq b$ holds in $D$, and *ko* otherwise. For instance, if $q(c,a,c)$ holds in $D$, then $\mathcal{M}(D,\{I\},U) = ko$.

Example 2 illustrates that the amount of violated cases never increases by any conservative update checked by an inconsistency-tolerant method. It also shows that it may decrease, since the violated case $\leftarrow p(a,a)$ may disappear after $p$ has been modified. Thus, it illustrates that such methods support *inconsistency-tolerant repairs*, i.e., updates that reduce the amount of violated cases, while surviving violations are tolerated.

By def. 2*a*), each constraint $I$ is a case of itself. Hence, each sound (3), resp., complete (4) inconsistency-tolerant method is sound (1) and, resp., complete (2). Thus, inconsistency-tolerant integrity checking significantly generalizes the traditional approach. The total integrity premise deprives the latter of a theoretical foundation, since integrity checking often is applied in the presence of inconsistency (which sometimes may not even be noticed). As opposed to that, inconsistency-tolerant methods tolerate any amount of inconsistency, including unsatisfiable integrity theories (cf subsection 4.4), while ensuring that conservative updates do not increase the amount of violated cases of integrity.

## 4 Inconsistency-Tolerant Integrity Maintenance

In 4.1, we show that methods for checking conservative updates can also be used for inconsistency-tolerant checks of changed integrity theories. In 4.2, we show how such checks can be made more efficient for dynamic schema maintenance. In 4.3, we propose a flexible inconsistency-tolerant dynamics for schema updates in safety-critical applications. In 4.4, we illustrate that inconsistency-tolerant schema maintenance can even tolerate unsatisfiable integrity theories.

### 4.1 Inconsistency-Tolerant Updates of Constraints

Each integrity checking method can be extended to deal with updates of the integrity theory, by resorting to brute-force checking: inserted constraints are plainly evaluated against the new database, without any simplification, while deleted constraints are simply dropped and ignored.

All methods that are inconsistency-tolerant for conservative updates are not known yet to be inconsistency-tolerant also for updates involving changes of the integrity theory. Theorem 1 fills that void.

**Theorem 1.** Each integrity checking method $\mathcal{M}$ that is inconsistency-tolerant for conservative updates and that, for arbitrary schema updates, checks new constraints brute-force, is inconsistency-tolerant for arbitrary updates.

*Proof.* We have to show that (3) holds for each $C \in \mathsf{SatCas}(D, IC^U \setminus IC)$. For each such $C$, there is an $I \in IC^U \setminus IC$ such that $C$ is a case of $I$. Now, let $\mathcal{M}(D,U) = ok$.

Thus, we have to show that $D(C) = sat$. Since $C$ is a case of $I$ and $I$ is inserted by $U$, $\mathcal{M}$ checks $I$ brute-force. It follows from $\mathcal{M}(D, U) = ok$ that $D^U(I) = sat$ (if $I$ would be violated in $D^U$, then, by definition, $\mathcal{M}$ would output $ko$). Since $I$ is satisfied if and only if all of its cases are satisfied, the result follows. □

*Example 3.* Suppose that the integrity theory in example 1 only contains the primary key constraint $I$. Further, consider an update $U'$ which consists of $U$ as in example 1 and additionally contains the insertion of $I'$. Then, each method $\mathcal{M}$ for efficient integrity checking known to this author will first decide that $I$ is not relevant for the insertion of $q(a, b)$, while tolerating any extant violation of $I$. Then, $\mathcal{M}$ will evaluate $I'$ against $D^U$, which leads to the output $ok$ if $D(I') = sat$ and $ko$ if not. The latter is the case if, e.g., $(c, c)$ is a row in $q$ and no row in $p$ matches $(c, z)$.

### 4.2 Improving Inconsistency-Tolerant Integrity Maintenance

Let us look at the following difference of examples 1 and 3. If $(c, c)$ is a row in $q$ and no row in $p$ matches $(c, z)$, then the violated case $\exists z(q(c, c) \rightarrow p(c, z))$ is tolerated in example 1, but not in example 3. Of course, it is sound that an update $U$ such as the one in example 3 is rebuked, and perhaps even desirable, since $U$ would increase the amount of violations in the database.

However, in a dynamic database that is operational while undergoing schema changes, it may be undesirable to delay operations until inserted constraints are evaluated entirely against possibly huge extensions of tables. Rather, it may be more desirable to just check some relevant cases, and, if they pass, have new integrity constraints be part of the updated schema, in order to prevent the current and all future updates from introducing new violated cases, at the expense of having to tolerante any extant violations of inserted constraints. If those exist at all, they can be dealt with at less busy times, e.g., in night runs.

Thus, a more efficient way to check each constraint $I$ inserted by some update $U$ is to focus on the cases of $I$ that are relevant for $U_c$, although $I$ is not part of the old database. After all, def. 2*d*) requires just that: (3) and (4) must hold for each case in $\mathsf{SatCas}(D, IC^U)$, where $D$ is the old database and $IC^U$ is the new integrity theory. Hence, it suffices to focus on relevant cases of inserted constraints. The following result identifies another advantage of this policy: it may achieve completeness, while brute-force checking of inserted constraints cannot.

**Theorem 2**
***a)*** No method $\mathcal{M}$ that checks inserted constraints brute-force is complete wrt inconsistency tolerance.
***b)*** Each method $\mathcal{M}$ that is sound and complete wrt inconsistency tolerance for conservative updates, and that, for each schema update $U$, focuses on cases of inserted constraints that are relevant for $U_c$, is sound and complete wrt inconsistency tolerance.

*Proof.* Part $a$) can be shown by the database $D$ and the update $U$ in example 3: if, for instance, $p(b, b)$ and $q(c, c)$ are the only facts about $p$ and $q$ in $D$, then, for each $\mathcal{M}$ that evaluates $I'$ brute-force, $\mathcal{M}(D, U) = ko$ holds. However, condition

(4) in definition 2d) warrants the output *ok*, since $D^U(C) = sat$ obviously holds for all $C \in \mathsf{SatCas}(D, IC^U)$. Hence, $\mathcal{M}$ is not complete. □

Part *b*) follows straightforwardly by verifying (3) and (4) by induction on the number of constraints inserted by $U$. □

*Example 4. (ex. 3 continued)* If $\mathcal{M}$ checks the insertion of $I'$ by focusing on its relevant cases for $U_c = insert\ q(a, b)$, then $\mathcal{M}$ outputs the same as in example 1. In particular, potential violations of the case $\exists z(q(c, c) \rightarrow p(c, z))$ are tolerated.

A striking advantage of checking inserted constraints just like all others, instead of checking them brute-force, is that the former is much more efficient than the latter. The latter is advocated in the SQL99 standard (cf. [15]), while the former lacks standardization but is common practice in most commercial systems, since it is much easier to use for dynamic schema maintenance than time-consuming brute-force checks. So, this paper can be understood as a contribution to capturing more precisely what is happening in practice.

### 4.3 Updating Safety-Critical Integrity Theories

The theme of this paper is inconsistency tolerance. Yet, the need of safety-critical applications to have a set of 'hard' integrity constraints totally satisfied at all times should not be lightheartedly compromised. Hence, methods are called for that can guarantee total satisfaction of all hard constraints for dynamic schema maintenance. Fortunately, each inconsistency-tolerant method is capable of providing such a service. To see this, we first define the following reliability property, then infer theorem 3 from it, and then interpret the definition and the result in terms of a dynamic maintenance of safety-critical applications.

**Definition 3.** $\mathcal{M}$ is called *reliable for hard constraints* if, for each pair $(D, U)$ and each integrity theory $IC_h$ such that $IC_h \subset IC$ and $D(IC_h) = sat,$ the following holds.

$$\text{If } \mathcal{M}(D, U) = ok \quad \text{then} \quad D^U(IC_h) = sat. \tag{5}$$

**Theorem 3.** Each inconsistency-tolerant $\mathcal{M}$ is reliable for hard constraints. □

By definition, each method that is reliable for hard constraints is capable of maintaining total satisfaction of a set $IC_h$ of hard constraints across updates, where $IC_h$ is a subset of the integrity theory $IC$ in the schema, even in the presence of violated constraints that are not hard. It is easy to see that (5) follows from (3). However, for methods that are not inconsistency-tolerant, such as the one in [7], (5) may not hold. Example 5 shows this.

*Example 5.* Let $IC = \{\leftarrow q(a), r(x, x),\ \leftarrow q(b), r(x, x)\}$, $IC_h = \{\leftarrow q(b), r(x, x)\}$, and $q(a)$, $r(b, b)$ be the only facts in $D$. Clearly, the case $\leftarrow q(a), r(b, b)$ is violated in $D$, while $D(IC_h) = sat$. Let $U = insert\ q(b)$. The only relevant case is $\leftarrow q(b), r(x, x)$. To check it, the method in [7] drops $q(b)$ since $U$ makes it *true*, thus obtaining the simplification $\leftarrow r(x, x)$. Since $\leftarrow q(a), r(x, x)$ is not relevant, the total integrity premise wrongly entails that $D^U(\leftarrow q(a), r(x, x)) = sat$. Now, assume that $q$ is locally accessible, while $r$ is remote and inaccessible. Thus, the method in [7] infers that also $D^U(\leftarrow r(x,x)) = sat$, since $q(a)$ is *true* in $D^U$. Hence, it unreliably outputs *ok*, although $D^U(IC_h) = vio$. As opposed to that, $\mathcal{M}(D, IC, U) = ko$ holds for each inconsistency-tolerant method $\mathcal{M}$. $\square$

### 4.4 Unsatisfiability

For each update, the use of an inconsistency-tolerant method to check integrity guarantees that all cases satisfied in the old state will remain satisfied in the new state. In fact, that continues to hold even if the given schema is unsatisfiable. Thus, inconsistency-tolerant methods are also unsatisfiability-tolerant. Hence, also the standard premise that the schema be satisfiable can be waived.

*Example 6.* Let the schema of $D$ consist of the rules $p(x,y) \leftarrow q(x,y), q(y,z)$, $q(x,y) \leftarrow r(x,y)$, $q(x,y) \leftarrow s(x,y)$ and $IC = \{\leftarrow p(x,x),\ \exists x,y\ r(x,y)\}$. Clearly, the schema is satisfiable. Let $(a,a)$, $(a,b)$ be all tuples in $s$. Thus, $\leftarrow p(a,a)$ is a violated case. There are no more violated ground cases of $\leftarrow p(x,x)$ if and only if neither $(b,a)$ nor any tuple matching $(x,x)$ is in $r$. Moreover, $\exists x,y\ r(x,y)$ is a violated case if and only if the extension of $r$ is empty. Now, let $U$ be the insertion of $q(x,y) \leftarrow r(y,x)$. Clearly, $U$ makes the schema unsatisfiable. However, each inconsistency-tolerant method $\mathcal{M}$ can be soundly applied to check $U$ for preserving satisfied cases. It is easy to see that $\mathcal{M}(D,U) = ko$ if and only if there is any tuple of form $(A,B)$ in $r$ such that $(B,A)$ is not in $r$ and $(A,B)$ does not match $(b,a)$ nor $(x,x)$. Otherwise, all cases of $\leftarrow p(x,x)$ that are violated in $D^U$ are already violated in $D$, hence $\mathcal{M}(D,U) = ok$.

Although, in example 6, the updated schema is unsatisfiable, it makes sense to accept further updates of $D^U$ that do not violate any satisfied case. Such updates may even lower the number of violated cases, e.g., $U' = delete\ s(a,a)$.

## 5 Conclusion

We have outlined how to extend traditional approaches to dynamic database change management that rely on established integrity checking methods. We have dealt with schema updates that may involve changes of the integrity theory, or make it unsatisfiable. In particular, we have shown that such updates can be dealt with efficiently and reliably, without compromising hard integrity requirements for safety-critical applications. Also, we have shown that the usual premises requiring the total satisfaction of each database state and the satisfiability of the database schema can simply be waived, for most methods.

Our definitions are as general as to cover arbitrary simultaneous changes of the schema and its extension. However, they are also as abstract as to define away or blind out other issues. In particular, we have not dealt with problems associated to the related issues of view updating (i.e., the translation of view update requests to updates of base relations), schema evolution (e.g., versioning, schema mergin and mapping, change propagation), transaction handling (e.g., to guarantee ACID properties) and distribution management. We intend to study these issues in future work. Currently, we are studying the application of inconsistency tolerance to concurrent transactions in replicated databases.

## References


1. Rodríguez, A.: Inconsistency Issues in Spatial Databases. In: Bertossi, L., Hunter, A., Schaub, T. (eds.) Inconsistency Tolerance. LNCS, vol. 3300, pp. 237–269. Springer, Heidelberg (2005)
2. Bry, F., Decker, H., Manthey, R.: A uniform approach to constraint satisfaction and constraint satisfiability in deductive databases. In: Schmidt, J.W., Missikoff, M., Ceri, S. (eds.) EDBT 1988. LNCS, vol. 303, pp. 488–505. Springer, Heidelberg (1988)
3. Cabot, J., Conesa, J.: Automatic Integrity Constraint Evolution due to Model Subtract Operations. In: Wang, S., Tanaka, K., Zhou, S., Ling, T.-W., Guan, J., Yang, D.-q., Grandi, F., Mangina, E.E., Song, I.-Y., Mayr, H.C. (eds.) ER Workshops 2004. LNCS, vol. 3289, pp. 350–362. Springer, Heidelberg (2004)
4. Christiansen, H., Martinenghi, D.: On simplification of database integrity constraints. Fundam. Inform. 71(4), 371–417 (2006)
5. Decker, H., Martinenghi, D.: A relaxed approach to integrity and inconsistency in databases. In: Hermann, M., Voronkov, A. (eds.) LPAR 2006. LNCS (LNAI), vol. 4246, pp. 287–301. Springer, Heidelberg (2006)
6. Decker, H., Martinenghi, D.: Classifying Integrity Checking Methods with regard to Inconsistency Tolerance. In: 10th PPDP. ACM Press, New York (2008)
7. Gupta, A., Sagiv, Y., Ullman, J., Widom, J.: Constraint checking with partial information. In: 13th PODS, pp. 45–55. ACM Press, New York (1994)
8. Lee, S.Y., Ling, T.W.: Further improvements on integrity constraint checking for stratifiable deductive databases. In: 22nd VLDB, pp. 495–505. Morgan Kaufmann, San Francisco (1996)
9. Lloyd, J., Sonenberg, L., Topor, R.: Integrity constraint checking in stratified databases. J. Log. Progr. 4(4), 331–343 (1987)
10. Martinenghi, D., Christiansen, H., Decker, H.: Integrity checking and maintenance in relational and deductive databases and beyond. In: Ma, Z. (ed.) Intelligent Databases: Technologies and Applications, pp. 238–285. Idea Group (2006)
11. Nicolas, J.M.: Logic for improving integrity checking in relational data bases. Acta Informatica 18, 227–253 (1982)
12. Rahm, E., Bernstein, P.: An online bibliography on schema evolution. SIGMOD Record 35(4), 30–31 (2006)
13. Ramakrishnan, R., Gehrke, J.: Database Management Systems. McGraw-Hill, New York (2003)
14. Sadri, F., Kowalski, R.: A theorem-proving approach to database integrity. In: Minker, J. (ed.) Foundations of Deductive Databases and Logic Programming, pp. 313–362. Morgan Kaufmann, San Francisco (1988)
15. Türker, C.: Schema Evolution in SQL-99 and Commercial (Object-)Relational DBMS. In: Balsters, H., De Brock, B., Conrad, S. (eds.) FoMLaDO 2000. LNCS, vol. 2065, pp. 1–32. Springer, Heidelberg (2001)
16. Wijsen, J.: Inconsistency and Incompleteness in Databases. In: Grust, T., Höpfner, H., Illarramendi, A., Jablonski, S., Mesiti, M., Müller, S., Patranjan, P.-L., Sattler, K.-U., Spiliopoulou, M., Wijsen, J. (eds.) EDBT 2006. LNCS, vol. 4254, pp. 278–374. Springer, Heidelberg (2006)

# Preface to FP-UML 2008

Juan Trujillo[1] and Andreas L. Opdahl[2]

[1] University of Alicante, Spain
[2] University of Bergen, Norway

The Unified Modeling Language (UML) has been widely accepted as the standard object-oriented (OO) modeling language for modeling various aspects of software and information systems. The UML is an extensible language, in the sense that it provides mechanisms to introduce new elements for specific domains if necessary, such as web applications, database applications, business modeling, software development processes, data warehouses. Furthermore, the latest version of UML 2.0 got even bigger and more complicated with more diagrams for some good reasons. Although UML provides different diagrams for modeling different aspects of a software system, not all of them need to be applied in most cases. Therefore, heuristics, design guidelines, lessons learned from experiences are extremely important for the effective use of UML 2.0 and to avoid unnecessary complication. Also, approaches are needed to better manage UML 2.0 and its extensions so they do not become too complex too manage in the end. Already, the many UML extensions are not well integrated and the UML 2.0 metamodel has become very complex.

The Fourth International Workshop on Foundations and Practices of UML (FP-UML'08) intends to be a a sequel to the successful BP-UML'05, BP-UML'06 and FP-UML'07, workshops held in conjunction with the ER'05, ER'06 and ER'07 respectively. FP-UML'08 intends to be an international forum for exchanging ideas on the best and new practices of the UML in modeling and system developments. Papers focused on the application on the UML in new domains and new experiences with UML 2.0, and foundations, theory and UML 2.0 extensions were also highly encouraged. As UML 2.0 is oriented towards the software design driven by models, papers applying the Model Driven Architecture (MDA) or the Model Driven Engineering (MDE) to specific domains were also highly encouraged. We hope the workshop be a forum for researchers, analyzers, designers, and users who use the UML to develop systems and software.

The workshop attracted papers from 7 different countries distributed all over the world: Chile, China, France, Pakistan, Spain, Tunisia and USA. We received 15 abstracts and 12 papers were finally submitted. The Program Committee only selected 5 papers, making an acceptance rate of 41.6%.

The accepted papers were organized in two sessions. In the first one, the two papers rely on aspects related to model transformations and Model Driven Architecture (MDA). In the second session, one paper presents a profile for modeling measurable requirements and the other two papers focus on specifying user requirements. Furthermore, this year we have a keynote entitled *"Using*

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 99–100, 2008.

*Object Concepts and UML for Conceptual Modeling"* given by Prof. Yair Wand from *The University of British Columbia, Vancouver, BC, Canada*.

We would like to express our gratitude to the program committee members and the external referees for their hard work in reviewing papers, the authors for submitting their papers, Prof. Yair Wand for accepting being the keynote speaker of FP-UML 2008 and the ER 2008 organizing committee for all their support.

## Organization

FP-UML'08 was organized within the framework of the following projects: the ESPIA project (TIN2007-67078) from the Spanish Ministry of Education and Science, and by the QUASIMODO project (PAC08-0157-0668) from the Castilla-La Mancha Ministry of Education and Science (Spain), and from the CALIPSO quality network (TIN2005-24055-E) from the Spanish Ministry of Education and Science.

# Using Object Concepts and UML for Conceptual Modeling

Yair Wand

Sauder School of Business
The University of British Columbia, Vancouver, BC, Canada
Yair.Wand@sauder.ubc.ca

**Abstract.** Developing an understanding of the application domain is a critical aspect of system development. A conceptual model is a representation of an application domain intended to support the analysis, documentation and communication of knowledge about the domain. However, despite their importance in acquiring and documenting this knowledge, conceptual modeling methods often do not provide well-formalized ways to create domain descriptions. The lack of formalization can lead to unclear, ambiguous, or incomplete models, as well as to variability of models of the same domain created by different individuals. In contrast, for other development activities - design and software modeling - highly formalized design approaches and modeling languages exist, notably the object-oriented approach and UML.

The introduction and wide acceptance of the object paradigm for system design has led to various efforts to adapt object concepts for creating representations of application domains. If successful, this would entail two benefits. First, the use of a well-formalized paradigm can lead to well-formalized conceptual modeling techniques. Second, similar concepts would then be used in various activities of system development - conceptual analysis, design, and coding. In this context, it has been claimed that objects might provide a "natural" way to model application domains, as humans think about the world in terms of discrete and dynamic units, akin to objects. All this leads to the question - how can object-based concepts and design languages, such as UML, be used in conceptual modeling. In this context, a modeling method comprises a modeling grammar (language) and a modeling process, which in turn includes guidelines and rules for mapping domain knowledge into the modeling grammar constructs.

We describe a long-term research effort on using object-concepts and UML for conceptual modeling. The first issue to be addressed was - providing "real world" meaning to object concepts. The approach towards accomplishing this was to adapt an ontological model of the world as a source of concepts to describe what exists and can happen. Then, a mapping was created between the ontological concepts and commonly used concepts such as objects, object properties, object classes, objects association, objects communication, and object composition. This mapping attach well-defined domain semantics to some object concepts, and provide a distinction between implementation aspects and application domain aspects of the object oriented approach. A second step involved

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 101–102, 2008.

the development of modeling rules to support the conceptual modeling process. The rules provide guidance for mapping the modeled domain into object concepts. As well, ontological premises provide semantic modeling rules - namely, rules that limit the “statements” that can be made in the model to those that make “ontological sense”. The rules can guide a structured modeling process and are independent of any specific modeling technique (graphical or otherwise). However, conceptual models are usually documented graphically. Hence, there is still a need for creating conceptual representations using a well defined modeling grammar with associated graphical notation. Thus, the next step was to explore the use of UML as the modeling grammar to support object-based conceptual modeling. This has led to a set of modeling rules that can be used when applying UML in creating conceptual models. The rules provide for determining which UML constructs should be used and which constructs should not be used to model specific types of domain phenomena; for how to combine UML constructs in ways that make “semantic sense”; and for intra- and inter- diagram integrity rules. For example, the rules prescribe specific links between class diagrams and state diagrams.

Clearly, the only way to test the approach is by empirical methods. We will conclude by describing some empirical results and practical experience, and by noting possible applications of the approach and further theoretical and empirical work that can be conducted.

# Towards Obtaining Analysis-Level Class and Use Case Diagrams from Business Process Models

Alfonso Rodríguez[1], Eduardo Fernández-Medina[2], and Mario Piattini[2]

[1] Computer Science and Information Technology Department
University of Bio-Bio,
Chillán, Chile
alfonso@ubiobio.cl
[2] ALARCOS Research Group, Information Systems and Technologies Department,
UCLM-Indra Research and Development Institute,
University of Castilla-La Mancha
Ciudad Real, Spain
{Eduardo.FdezMedina,Mario.Piattini}@uclm.es

**Abstract.** Nowadays, business process modeling, using industrial standards such as UML or BPMN, offers us a good opportunity to incorporate requirements at high levels of abstraction. In the context of Model Driven Architecture (MDA), the business process model is considered as a Computation Independent Model (CIM). In our proposal we will transform the business process specifications into analysis-level classes and use cases, both of which are UML artifacts used to describe the problem in the context of Platform Independent Models (PIM). Such artifacts are complementary, as they are only a subset of the analysis-level classes and use cases that describe the whole problem, in the first stages of the software development process. This work contains the principle issues involved in the main standards that allow us to represent a business process, details of the transformation rules in QVT specification and an illustrative example in which our proposal has been applied.

## 1 Introduction

A Business Process (BP) is the combination of a set of activities within an enterprise with a structure describing their logical order and dependence whose objective is to produce a desired result. A model is a simplified view of a complex reality. It is a means of creating abstraction, thus allowing one to eliminate irrelevant details and focus on one or more important aspects at a time. Business process models enable a common understanding and facilitate discussion among different stakeholders in the business [1, 9].

Furthermore, requirement specification usually results in a specification of the software system which should be as exact as possible [2], since effective business process models facilitate discussion between different stakeholders in the business, allowing them to agree on the key fundamentals and to work towards common goals [9].

Several languages and notations exist for business process modeling [10] and of these, the Unified Modeling Language (UML) and the Business Process Modeling Notation (BPMN) are widely accepted standard notations [15].

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 103–112, 2008.

The Model Driven Architecture (MDA) approach [16] is not based on one single idea. Among its objectives are: the separation of business-neutral descriptions and platform dependent implementations, the expression of specific aspects of a system under development with specialized domain-specific languages, the establishment of precise relationships between these different languages in a global framework and, in particular, the capability of expressing operational transformations between them [4]. For the transformation of models, OMG has proposed Query/View/Transformation (QVT) [18], in order to seek an answer which is compatible with its MDA standard suite: UML, MOF, OCL, etc [12].

Our proposal is within the MDA scope, and we have used the QVT transformation to move from a business process model (CIM) to analysis-level classes and use cases (PIM). The artifacts obtained can complement the requirements captured in a software development process. For this purpose, we have chosen the UP (Unified Process) [11], which is composed of a set of activities necessary for transforming users' requirements into a software system, due to the fact that it is a consolidated and successful software construction method.

The structure of the rest of the paper is as follows: in Section 2, we shall summarize the main issues in relation to business processes modeling. In Section 3, we shall present CIM to PIM transformations, for analysis-level classes as well as use cases. Finally, in Section 4, we shall present an illustrative example and in Section 5 our conclusions will be drawn.

## 2 Business Processes Modeling

In business process modeling, the main objective is to produce a description of a reality (for example, the way in which a commercial transaction is carried out) to understand and eventually modify it with the aim of incorporating improvements into it. As a consequence, it is important to have a notation that allows us to model the essence of the business as clearly as possible. This notation must allow us to incorporate different perspectives which give place to different diagrams in which the rules, goals and objectives of the business, and not only relationships but also interactions, are shown [7]. A significant part of the success of modeling has to do with its ability to express the different needs of the business as well as its having a notation in which these needs can be described. This is why, when choosing an approach and/or notation, the properties of the object to be modelled must be taken into account, in other words, the business process, the environmental features and the underlying reasons for its use [5].

Among those techniques that have been used for business process modeling are the following: flow diagrams, data flow diagrams, entity-relationship diagrams, state-transition diagrams, Gantt charts, Role Activity Diagrams (RAD), the family of techniques known as Integration Definition for Function Modeling (IDEF), Petri Nets, simulation, techniques based on knowledge (artificial intelligence) and workflow techniques [1, 10]. At present, and according to the state of the business process modeling industry [15], it is possible to identify UML [17] and BPMN [6] among the main standards.

In UML 2.0 the element used to represented business process and workflows is the Activity Diagrams (UML 2.0-AD) [13]. In previous UML versions, expressivity was limited and this fact confused users who did not use orientation towards objects as an approach for modeling. However, it is now possible to support flow modeling across a wide variety of domains. The UML 2.0-AD elements that will be used in our proposal are Activity Partition, Action, Data Store Node, and Interruptible Activity Region.

In BPMN the elements used to process representation is the Business Process Diagram (BPMN-BPD). This diagram was designed to facilitate its use and understanding, and to offer an expressive force with which to model complex businesses. The BPMN-BPD elements that will be used in our proposal are Pool, Lane, Data Objects, Group, and Activity.

## 3 CIM2PIM Mappings

In our opinion, a business process which has been built by a business analyst is not only useful in the specific business field, but is also very useful in a software construction process. From this process we can obtain system requirements, a stage taken into account by all modern development processes that basically consists of obtaining from the customer or the interested parties the system requirements for developing software construction from this point. In our proposal, CIM2PIM transformations are aimed at obtaining useful artifacts in software development. Both the analysis-level classes and the use cases obtained from the business process model become part of an ordered and systematic process of software development.

A fundamental aspect MDA models transformation. The Object Management Group (OMG) proposal which allows us to perform this task is Query/View/ Transformation (QVT) [18]. QVT is compatible with the MDA standard since its abstract syntax is defined as a MOF 2.0 (Meta Object Facility) metamodel. Basically, QVT offers us the possibility of manipulating models by considering queries that use a model as an input. It selects specific elements of this model according to a search pattern, views corresponding to models that are derived from other models, and finally transformations, which use one or more input models to obtain an output model or result as a reference.

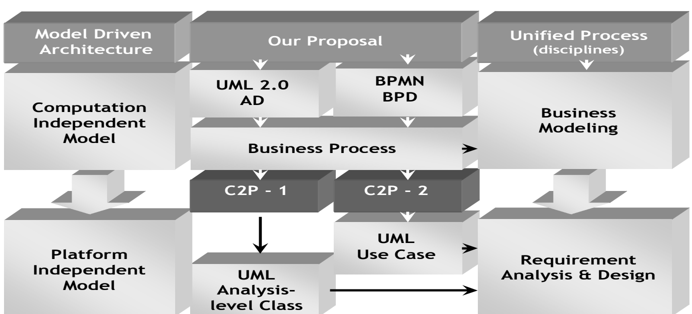



**Fig. 1.** Our proposal overview

In Figure 1, the basic aspects of our proposal are shown. The BP model can be specified with UML 2.0-AD or BPMN-BPD. In an MDA approach, such a description corresponds to a computation independent model. Through the application of a set of transformation rules described with QVT (C2P-1 y C2P-2) it is possible to obtain a subset of the analysis-level classes and use cases that facilitate the understanding of the problem. UP is considered in our proposal because both the BP description and the artifacts obtained through the transformation can be used during the first disciplines. Thus, the business description will be useful in the "Business Modeling" discipline and both analysis-level classes and use cases complement the "Requirement" and "Analysis & Design" disciplines.

The set of transformation rules that we propose consider as an input a BP model described with UML 2.0-AD. In order to include models described with BPMN-BPD, we have established an equivalence relationship between the elements of both notations. Although the relationship between these two notations is dealt with in [22], the equivalents that we have proposed permit a relationship to exist between the concepts which represent similar business process concepts. These equivalence relationships have been expressed using QVT language (see Table 1).

**Table 1.** Mapping between BPMN-BPD and UML 2.0-AD elements

```
transformation BPMN-BPD2UML-AD
  top relation R1   // from Pool to Activity Partition
  {
    checkonly domain bpmn_BusinessProcessDiagram p:Pool {name = n}
    enforce domain uml_ActivityDiagram ap:ActivityPartition {name = n}
  }
  top relation R2   // from Lane to Activity Partition
  {
    checkonly domain bpmn_BusinessProcessDiagram l:Lane {name = n}
    enforce domain uml_ActivityDiagram ap:ActivityPartition {name = n}
  }
  top relation R3   // from Group to Interruptible Activity Region
  {
    checkonly domain bpmn_BusinessProcessDiagram g:Group {name = n}
    enforce domain uml_ActivityDiagram ir:InterruptibleActivityRegion {name = n}
  }
  top relation R4   // from Activity to Action
  {
    checkonly domain bpmn_BusinessProcessDiagram ac:Activity {name = n}
    enforce domain uml_ActivityDiagram act:Action {name = n}
  }
  top relation R5   // from Data Object to Data Store Node
  {
    checkonly domain bpmn_BusinessProcessDiagram do:DataObject {name = n}
    enforce domain uml_ActivityDiagram dsn:DataStoreNode {name = n}
  }
  top relationR6   // from Start Event to Initial Node
  {
    checkonly domain bpmn_BusinessProcessDiagram s:StarEvent {name = n}
    enforce domain uml_ActivityDiagram act:Action {name = n}
  }
```

In sections 3.1 and 3.2, we have given a detail description of the transformations from a Business Process to analysis-level classes and use cases respectively. In each case, the set of rules and their specification in QVT text form will be shown.

### 3.1 Business Process to Analysis-Level Class (C2P-1)

In this section, we will present the set of rules that allows us to obtain analysis-level classes from a business process specification.

In our review of the literature related to this subject, only two works dealing directly with this type of transformations were found. In the first [3], activity diagrams are transformed into analysis classes. This transformation is not automatically performed and a previous version of UML 2.0 is used. In the second work [20], the software designer studies the BP model described with BPMN by extracting the UML classes which are later refined. The difference between these proposals and ours is that, in both cases, the generation process of analysis classes is manually performed and the result of transformations is not related to a software development process.

The transformation from a business process model specified with UML 2.0-AD (or its equivalence in BPMN-BPD) to analysis-level classes are described with QVT language in Table 2.

**Table 2.** Mapping between Activity Diagrams and Class Diagrams elements

```
transformation ActivityDiagram2ClassDiagram
  top relation R1   // from Activity Partition to Class
  {
    checkonly domain uml_ActivityDiagram ap:ActivityPartition {name = n}
    enforce domain uml_ClassDiagram c:Class {name = n}
    where {
          ap.containedNode → forAll(cn:Action|R4(cn))
    }
  }
  top relation R2   // from Interruptible Activity Region to Class
  {
    checkonly domain uml_ActivityDiagram iar:InterruptibleActivityRegion {name = n}
    enforce domain uml_ClassDiagram c:Class {name = n}
  }
  top relation R3   // from Data Store Node to Class
  {
    checkonly domain uml_ActivityDiagram dsn:DataStoreNode {name = n}
    enforce domain uml_ClassDiagram c:Class {name = n}
  }
  relation R4   // from Action to Operation in Class
  {
    checkonly domain uml_ActivityDiagram ac:Action {name = n, inPartition=ap}
    enforce domain uml_ClassDiagram op:Operation {name = n, ownerClass=c:Class{name=ap.name}}
  }
```

Additionally, we present the set of rules that permits analysis-level class refinement (see Table 3). These rules are applied later than QVT rules. Their main objective is that of enriching the class model through the incorporation of meaningful region names, the identification of relationships between the classes obtained and the elimination of redundant elements.

**Table 3.** Refinement Rules for Analysis-Level Classes

| | |
|---|---|
| **RR 1:** | Region Name is obtained by linking the ActivityPartition names where the InterruptibleActivityRegion is contained |
| **RR 2:** | Composition relationships are obtained from top and middle ActivityPartitions |
| **RR 3:** | Redundant specifications must be eliminated |

### 3.2 Business Process to Use Case (C2P-2)

In this section, we will present the set of rules that permit us to obtain use cases from a business process description.

In our review of literature, we discovered that in [19], the possibility of manually obtaining use cases from a BP specification made with BPMN is suggested. In [14], the automatic attainment of UML artifacts from a BP description that was made using BPMNN is proposed. The authors extend the BPMN (Extension Level 1) in order to add information about the sequence and the input and output flows. This allows them to apply rules from which use cases, state diagrams, sequence and collaboration are attained. In [21], a manually performed transformation from a BP described with AD-UML 2.0 to use cases is stated and finally, in [8], use cases are obtained from business process models which are not represented by activity diagrams. The differences between these proposals and ours are basically the following: (i) even in works where there are automatic transformations, previous manual intervention is required, ii) transformations are not described by using languages which have been specially designed for this purpose and iii) the result of the transformations does not appear to be linked to a business process development.

The transformations from a business process model specified with AD-UML 2.0 (or its equivalence in BPD-BPMN) to use cases are carried out in accordance with the QVT rules described in Table 4.

**Table 4.** Mapping between Activity Diagrams and Use Case elements

```
transformation ActivityDiagram2UseCaseDiagram
  top relation R1   // from Activity Partition to Actor
  {
    checkonly domain uml_ActivityDiagram ap:ActivityPartition {name = n}
    enforce domain uml_UseCaseDiagram a:Actor{name = n}
    where {
         ap.containedNode → forAll(cn:Action|R3(cn))
    }
  }
  top relation R2   // from Interruptible Activity Region to Actor
  {
    checkonly domain uml_ActivityDiagram iar:InterruptibleActivityRegion {name = n}
    enforce domain uml_UseCaseDiagram a:Actor {name = n}
    where {
         ap.containedNode → forAll(cn:Action|R3(cn))
    }
  }
  relation R3   // from Action to UseCase
  {
    checkonly domain uml_ActivityDiagram ac:Action {name = n, inPartition=ap}
    enforce domain uml_UseCaseDiagram uc:UseCase {name = n, subject= ACTORS: Set(Actor)};
    where {
         ACTORS→including (a:Actor{name=ap.name})
    }
  }
```

**Table 5.** Refinement rules for Use Cases

| | |
|---|---|
| **RR 1:** | Subject name is obtained from the business process name |
| **RR 2:** | Region Name is obtained by linking the Activity Partition names where Interruptible Activity Region is contained |
| **RR 4:** | Main Actor corresponds to the Activity Partition or region name where Initial Node is present |
| **RR 5:** | Actor Generalization is obtained from top and middle Activity Partitions |
| **RR 6:** | Redundant specifications must be eliminated |

We also present a set of rules that permits use case refining (see Table 5). These rules are applied later than QVT rules. Their main objective is that of enriching the use case through the incorporation of the subject name, region names, identifying the main actor, actor generalization and the elimination of redundant elements.

## 4 Example

Our illustrative example (see Figure 2) describes a typical business process for the admission of patients to a health-care institution. In this case, the business analyst identified the following Activity Partitions: Patient, Administration Area (which is a top partition that is divided into the Admission and Accounting middle partitions), and the Medical Area (divided into Medical Evaluation and Examinations). We shall apply the transformations described in the previous section to this business process.

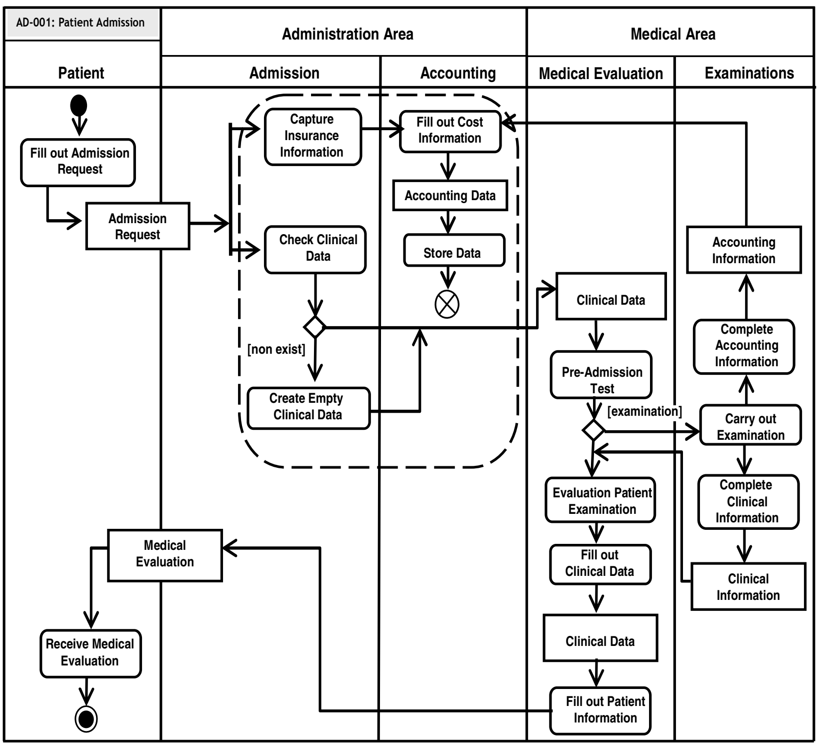



**Fig. 2.** Admission of Patients to a Medical Institution

The result of the application of the QVT and refinement rules for obtaining analysis-level classes and use case elements from the example is shown in Table 6.

**Table 6.** Mapping from UML 2.0-AD to analysis-level classes and use cases elements

| Rule | UML2.0-AD elements | To | Analysis-level class element |
|---|---|---|---|
| C2P-1/R1 | Activity Partition | Class | Patient, Administration Area, Admission, Accounting, Medical Area, Medical Evaluation, Examinations |
| C2P-1/R2 | Interruptible Activity Partition | Class | Region 01 (AdmissionAccounting) |
| C2P-1/R3 | Data Store Node | Class | Admission Request, Accounting Data, Clinical Data, Accounting Information, Clinical Information and Medical Evaluation |
| C2P-1/R4 | Action | Operation | Fill out Admission Request, Receive Medical Evaluation, Capture Insurance Information, Check Clinical Data, Create Empty Clinical Data, Fill out Cost Information, Store Data, Pre-Admission Test, Evaluate Patient Examinations, Fill out Clinical Data, Fill out Patient Information, Complete Accounting Information, Carry out Examinations and Complete Clinical information |
| C2P-2/R1 | Activity Partition | Actor | Patient, Administration Area, Admission, Accounting, Medical Area and Medical Evaluation and Examinations |
| C2P-2/R2 | Interruptible Activity Region | Actor | Region 01 (AdmissionAccounting) |
| C2P-2/R3 | Action | Use Case | Fill out Admission Request, Receive Medical Evaluation, Capture Insurance Information, Check Clinical Data, Create Empty Clinical Data, Fill out Cost Information, Store Data, Pre-Admission Test, Evaluate Patient Examinations, Fill out Clinical Data, Fill out Patient Information, Complete Accounting Information, Carry out Examinations and Complete Clinical information |

In Figure 3, analysis-level classes obtained from a business process specification for patient admission are graphically shown.

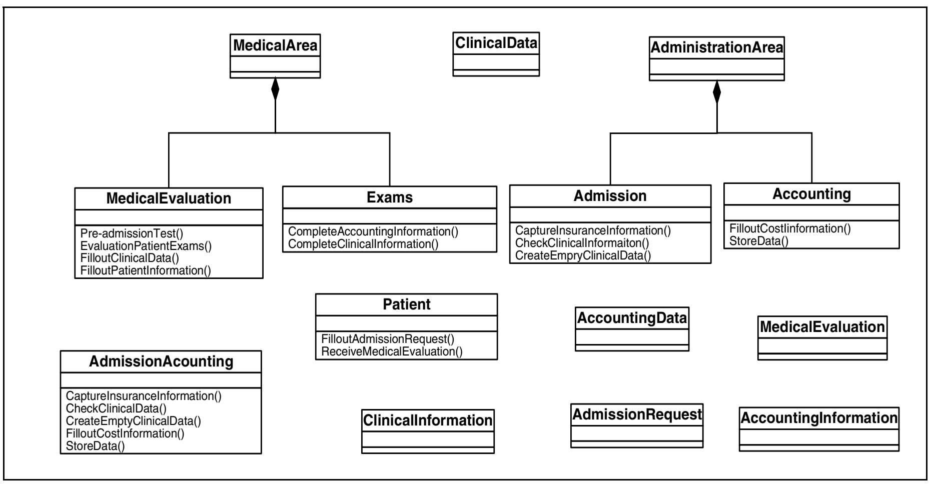



**Fig. 3.** Class Diagram from "Patient Admission" Business Process Specification

In Figure 4, the use case is graphically shown. This figure is enriched since, after the semi-automatic derivation, we have incorporated the use case identification, the main actor identification, the hierarchization of the areas that have subpartitions and the region denomination. In order to highlight these improvements, they have been marked with the symbol (*).

Not only analysis-level classes but also use cases complement UP at the "Requirement" and "Analysis & Design" disciplines. These artifacts form a subset of the total number of artifacts that will finally be necessary for software construction.

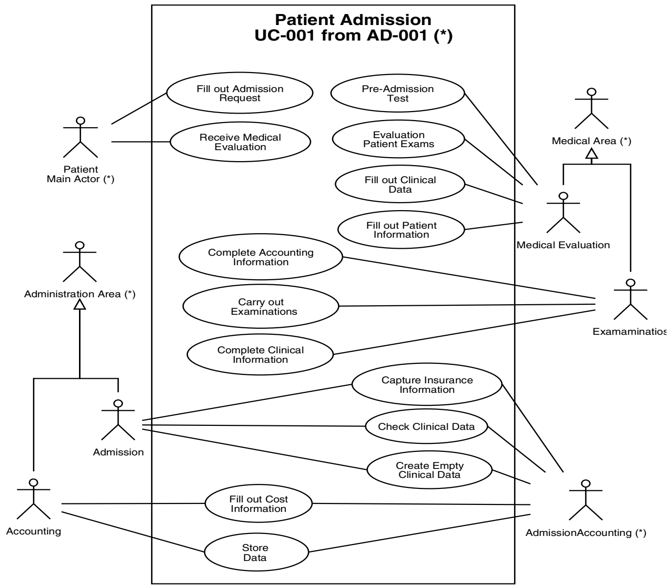



**Fig. 4.** Use case from "Patient Admission" Business Process Specification

## 5 Conclusions and Ongoing Work

The improvement presented by the UML 2.0 language and the appearance of the BPMN notation related to business process specification allow us to consider such specifications as a source of requirements to be used as an input in a software development process.

In this paper, we have presented CIM to PIM transformations in an MDA focused environment. A business process specification made by a business analyst (CIM) was used to obtain analysis-level classes and use cases (PIM). The transformation rules were specified with the use of QVT language. Both the analysis-level classes and the use cases can be used in a systematic and ordered software development process.

Ongoing work is oriented towards enriching transformations to make it possible to obtain more complete models of analysis-level classes and use cases. Our future work also has the purpose of optimizing the prototype that we have created to carry out the transformations with the aim of improving specification reuse and documentation.

**Acknowledgments.** This research is part of the following projects: DIMENSIONS (PBC-05-012-1), and MISTICO (PBC06-0082) both partiality supported by the FEDER and the "Consejería de Ciencia y Tecnología de la Junta de Comunidades de Castilla-La Mancha", Spain, COMPETISOFT (506PI287), granted by CYTED and ESFINGE (TIN2006-15175-C05-05/) granted by the "Dirección General de Investigación del Ministerio de Ciencia y Tecnología", Spain.

## References


1. Aguilar-Savén, R.S.: Business process modelling: Review and framework. International Journal of Production Economics 90(2), 129–149 (2004)
2. Artelsmair, C., Wagner, R.: Towards a Security Engineering Process. In: The 7th World Multiconference on Systemics, Cybernetics and Informatics, vol. VI, pp. 22–27 (2003)
3. Barros, J.P., Gomes, L.: From Activity Diagrams to Class Diagrams. In: Workshop Dynamic Behaviour in UML Models: Semantic Questions In conjunction with Third International Conference on UML (2000)
4. Bézivin, J.: In Search of a Basic Principle for Model Driven Engineering. UPGRADE, European Journal for the Informatics Professional V (2), 21–24 (2004)
5. Bider, I.: Choosing Approach to Business Process Modeling - Practical Perspective, `http://www.ibissoft.se/english/howto.pdf`
6. BPMN: Business Process Modeling Notation (BPMN) 04.pdf (2020), `http://www.bpmn.org/Documents/BPMN%20V1-0%20May%203%`
7. Castela, N., Tribolet, J., Silva, A., Guerra, A.: Business Process Modeling with UML. In: 3rd. International Conference on Enterprise Information Systems, vol. 2, pp. 679–685 (2001)
8. Dijkman, R.M., Joosten, S.M.M.: An Algorithm to Derive Use Cases from Business Processes. In: 6th International Conference on Software Engineering and Applications (SEA), pp. 679–684 (2002)
9. Eriksson, H.-E., Penker, M.: Business Modeling with UML. OMG Press (2001)
10. Giaglis, G.M.: A Taxonomy of Business Process Modelling and Information Systems Modelling Techniques. International Journal of Flexible Manufacturing Systems 13(2), 209–228 (2001)
11. Jacobson, I., Booch, G., Rumbaugh, J.: The Unified Software Development Process (1999)
12. Jouault, F., Kurtev, I.: On the architectural alignment of ATL and QVT. In: ACM Symposium on Applied Computing - Model Transformation, pp. 1188–1195 (2006)
13. Kalnins, A., Barzdins, J., Celms, E.: UML Business Modeling Profile. In: Thirteenth International Conference on Information Systems Development, Advances in Theory, Practice and Education, pp. 182–194 (2004)
14. Liew, P., Kontogiannis, P., Tong, T.: A Framework for Business Model Driven Development. In: $12^{th}$ International Workshop on Software Technology and Engineering Practice (STEP), pp. 47–56 (2004)
15. Lonjon, A.: Business Process Modeling and Standardization (2004), `http://www.bptrends.com/`
16. Object Management Group: MDA Guide Version 1.0.1, `http://www.omg.org/docs/omg/03-06-01.pdf`
17. OMG: Object Management Group, `http://www.omg.org/`
18. QVT: Meta Object Facility (MOF) 2.0 Query/View/Transformation Specification. OMG Adopted Specification ptc/05-11-01 (2005)
19. Rungworawut, W., Senivongse, T.: A Guideline to Mapping Business Processes to UML Class Diagrams. WSEAS Trans. on Computers 4(11), 1526–1533 (2005)
20. Rungworawut, W., Senivongse, T.: Using Ontology Search in the Design of Class Diagram from Business Process Model. Enformatika, Transactions on Engineering, Computing and Technology 12, 165–170 (2006)
21. Štolfa, S., Vondrák, I.: A Description of Business Process Modeling as a Tool for Definition of Requirements Specification. In: Systems Integration 12th Annual International Conference, pp. 463–469 (2004)
22. White, S.A.: Process Modeling Notations and Workflow Patterns, `http://www.ebpml.org/bpmn.htm`

# Improving Automatic UML2 Profile Generation for MDA Industrial Development*

Giovanni Giachetti, Francisco Valverde, and Oscar Pastor

Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
Camino de Vera s/n
46022 Valencia, España
{ggiachetti,fvalverde,opastor}@dsic.upv.es

**Abstract.** In the context of MDA-oriented solutions, a modeling language with a precise semantics is a mandatory requirement. Even though MDA encourages the use of UML, there are several MDA approaches that define their own Domain Specific Modeling Languages (DSML) in order to obtain the needed semantic precision. However, the high acceptance of UML in the software industry has led different MDA-approaches to integrate their DSMLs semantics into UML by means of UML profiles. Certain approaches provide alternatives to elaborate a UML profile from the DSML metamodel, but generally speaking, the resultant profile does not include all the semantic expressiveness of the original DSML. This paper provides a solution that is focused on the MDA industrial development. This solution is based on a systematic approach to generate a DSML metamodel that is used to automatically obtain a UML profile that has all the expressiveness of the original DSML.

**Keywords:** UML Profile, UML2, MDA, MDD, Model-driven Engineering, DSML, Metamodeling.

## 1 Introduction

The *Model Driven Architecture* (MDA) [6] suggests the use of UML to define the Conceptual Models that support a *Model Driven Development* (MDD) process. However, UML requires additional semantics to be able to describe in a precise way all the *conceptual constructs* related to a complete MDD process [3]. This lack of semantic precision can be observed in the *semantic extension points*, in which the UML specification presents multiple semantic interpretations for the same conceptual construct [12]. As a consequence, several MDA approaches have defined their *Domain Specific Modeling Languages (DSML)* in order to obtain the required semantic precision.

Nevertheless, for MDA approaches to be widely accepted by the software industry the use of UML is a key requirement, since it is the standard language for software modeling. For this reason, the MDA approaches (that have defined DSMLs) are looking for alternatives to use UML without losing the semantic precision of their DSMLs.

* This work has been developed with the support of MEC under the project SESAMO TIN2007-62894.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 113–122, 2008.

An interesting approach for solving the semantic gap between a DSML and UML is the definition of a UML profile [1] that is oriented to extend UML with the specific DSML semantics. The use of a UML profile is a recommended approach because of the availability of tool support and a more effective learning curve [1] [13].

Taking into account previous industrial MDA experiences [10], two main requirements must be fulfilled in order to apply a UML profile solution in an industrial MDA-based method:

- The UML profile must be easily adapted to the evolution of the MDA approach because MDD industrial approaches continuously change in order to introduce new features.
- The UML profile must provide all the semantic expressiveness and precision of the DSML in order to use UML to describe the needed conceptual models.

Related works indicate that the manual definition of a UML profile is not a suitable option because it is an error-prone and time-consuming task [14]. These two risk factors (time and errors) must be avoided in a MDD industrial context, where time costs money, and mistakes in implementation directly impact on customers satisfaction. Therefore, the automatic generation of a UML profile from a DSML is a desired solution to give support to the presented requirements.

Certain works related to UML profiles have defined proposals to achieve a semi-automated profile generation. These works consider the generation from the metamodel that describes the *conceptual constructs* related to a DSML, through a mapping between this *DSML metamodel* and the UML metamodel. However, as far as our knowledge, there is no approach that supports the two requirements mentioned above.

The objective of this work is to introduce a process to automatically generate a UML profile that satisfies the requirements of an industrial MDA process. To achieve this objective, the definition of a specific DSML metamodel, named the *Integration Metamodel*, is required. The Integration Metamodel is a DSML metamodel with a structure that allows a perfect integration with the UML metamodel (UML2 Superstructure). This metamodel includes all the mapping information required to automatically generate a UML profile with all the semantic expressiveness and precision that the original DSML has.

This paper is focused on the definition of the Integration Metamodel for automatic UML profile generation. Therefore, this research work: 1) explains why an Integration Metamodel is needed for this proposal; 2) defines a systematic approach to obtain this Integration Metamodel; and 3) shows its benefits for the automatic profile generation focused on industrial MDA approaches.

The rest of the paper is organized as follows: section 2 shows the background related to UML profile generation. Section 3 introduces the proposed process for the automatic generation of a UML profile, taking an Integration Metamodel definition as input. Section 4 introduces the Integration Metamodel and proposes a systematic approach to define it. Section 5 shows the benefits provided by the use of the Integration Metamodel. Finally, Section 6 presents our conclusions and further work.

## 2 Background

In the literature related to defining UML profiles to bridge UML with DSML, two main working schemata can be observed: 1) the definition of the UML profile from scratch; and 2) the definition of the UML profile starting from the DSML metamodel.

In this work, the second working schema has been selected, because it provides a methodological solution. One of the first proposals related to this working schema is the work presented by Fuentes-Fernández in [4], who proposes some basic guidelines for the UML profile definition. Selic in [12] proposes a systematic approach taking into account the new UML profile characteristics introduced in UML2. In addition, this approach establishes some guidelines to ensure a correct DSML metamodel definition and suggests how a correct mapping between the DSML metamodel and the UML metamodel can be defined. More recent works such as Lagarde in [5] and Wimmer in [14] present how the profile definition can be partially automated. Lagarde [5] proposes a systematic approach that can be partially automated through the identification of specific design patterns. However, this approach requires the manual definition of an initial profile skeleton. Wimmer [14] proposes a semi-automatic approach that introduces a specific mapping language that is oriented to define a mapping between the DSML metamodel and the UML metamodel. This mapping allows an automated UML profile generation. However, this approach only supports one to one (1:1) equivalences between the two metamodels: one element of the DSML metamodel is mapped to one element of the UML metamodel and vice versa. As a consequence, the effective application of this approach in an industrial MDA process is not possible because, in real MDA approaches, the mappings between the two metamodels may be different than 1:1 (for instance, 1:M, M:1, or even M:M). This issue is discussed in more detail in section 4.1.

Although, Bran Selic in [12] and other related works refer to the metamodel of a DSML as *domain model*, in this paper the term *DSML metamodel* is used to avoid confusions between the concepts of model and metamodel.

The works mentioned above have some limitations to achieve a completely automated generation. In general terms, these limitations come from the structural differences between the DSML metamodel and the UML metamodel. These differences prevent the definition of an adequate identification of equivalences between the two metamodels (pattern identification, mappings, etc.). This causes the resulting UML Profile to have different semantic expressiveness than the original DSML and also it makes automatic generation difficult.

## 3 A Process to Generate a UML Profile from a DSML

According to the background described above, the main problems in generating a UML profile from a DSML are the structural differences between the DSML metamodel and the UML metamodel. In this section, a UML profile generation process that solve these structural problems by means of the definition of an *Integration Metamodel* is introduced (Fig. 1).

The Integration Metamodel is a special DSML metamodel defined to automate the integration of a DSML into UML. The Integration Metamodel is defined from the

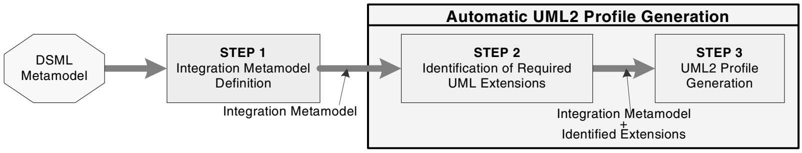



**Fig. 1.** Process to automate the profile generation

DSML metamodel and has all the semantics of the original metamodel. The only difference is the structure of the Integration Metamodel because it is defined to allow a perfect mapping with the UML Metamodel. This mapping information is included inside the Integration Metamodel definition. The UML metamodel considered for the definition of the Integration Metamodel is the UML2 Superstructure [8].

The process presented in Fig. 1 is composed of the following three steps:

- *Step 1***:** The definition of the Integration Metamodel taking into account the UML2 Superstructure [8].
- *Step 2***:** The identification of the required UML extensions. This identification is performed by means of a comparison between the Integration Metamodel and the UML2 Superstructure, using the structure of the Integration Metamodel and the mapping information.
- *Step 3***:** The generation of the UML2 Profile according to a set of transformation rules.

The second and third step can be automated in order to obtain an automatic UML profile generation process. In this paper, only the first step is presented due to space constraints and because this first step is the most relevant to achieve an automated process. The Integration Metamodel approach presented in this work, can be used to improve other proposals oriented to extend UML with the semantics of DSMLs.

# 4 The Integration Metamodel: Improving the Integration between DSMLs and UML2

The Integration Metamodel assures the generation of a UML Profile that provides all the semantic expressiveness involved in the original DSML. Thus, the requirements presented in the introduction are satisfied. This section shows how an Integration Metamodel, which provides an adequate mapping with the UML2 Superstructure, facilitates a correct UML profile generation. Next, a systematic approach that is oriented to obtaining the Integration Metamodel is presented, and an implementation schema for the Integration Metamodel is proposed. Finally, the benefits of the Integration Metamodel are briefly described.

## 4.1 The Need for a Correct Mapping with the UML2 Superstructure

In this section, a brief example is introduced to understand why a correct mapping is required to generate a UML profile. Fig. 2 shows this example.

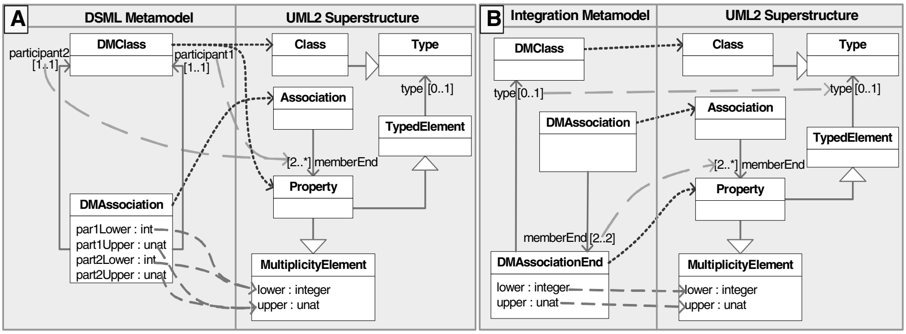



**Fig. 2.** Simplified mapping between a generic binary association and the UML2 Superstructure

Fig.2-A presents a DSML metamodel for generic binary association. The classes that participate in the association (association ends) are specified through the associations *participant1* and *participant2*. The cardinality for each association end is represented in the class *DMAssociation* by the attributes *part1Lower* and *part1Upper* for *participant1*, and the attributes *part2Lower* and *part2Upper* for *participant2*. Two issues related to the mapping performed in this figure (2-A) are analyzed below:

- The class *DMClass* is mapped to the UML classes *Class* and *Property*. The reason is because, in UML, the class *Property* identifies the classes that participate in the association (through the association *type* inherited from the class *TypedElement*). This mapping 1:2 (one for the DSML metamodel side and two for the UML2 Superstructure side) does not allow a correct stereotype to be defined. According to the works reviewed, this is because the UML profile generation involves the definition of a stereotype to represent each class (conceptual construct) of the DSML metamodel. In this case, the semantics and the properties related to each mapping are different, and a single stereotype cannot extend two UML metaclasses with different semantics and properties for each metaclass.
- The class *DMAssociation* presents a similar issue because the semantics of the class *DMAssociation* is distributed between two UML classes (*Association* and *MultiplicityElement*). In addition, the associations *participant1* and *participant2* of the class *DMAssociation* are represented by one association in the UML, and the type related to these associations is different in each metamodel (in the DSML metamodel the type corresponds to a class, while in the UML2 Superstructure the type corresponds to a property). This cannot be represented by a stereotype because the tagged values cannot redefine UML properties to change the related type.

The proposal to solve these mapping issues is to restructure the DSML metamodel to align it with the UML2 Superstructure, thereby obtaining the Integration Metamodel. Fig. 2-B shows this solution. The identification of problematic mappings is important to be able to perform a correct UML profile definition. However, performing a complex analysis for each defined mapping is not a suitable approach. In order to simplify this task, the identification can be performed through a set of constraints that are motivated by the mapping issues mentioned above. These constraints are:

- All the classes from the DSML metamodel must be mapped.
- The mapping must be defined between elements of the same type (classes with classes, attributes with attributes, and so on).
- An element from the DSML metamodel is only mapped to one element of the UML2 Superstructure. This is a M:1 mapping; for instance, many classes of the DSML metamodel can be mapped to one class of UML. Fig. 2-B shows that all the defined mappings are 1:1.
- If the properties (attributes and associations) of a class from the DSML metamodel are mapped to properties of a UML class, then the class that owns the properties is mapped to this UML class (or a generalization of it). To understand this second constraint, the class *DMAssociationEnd* (Fig. 2-B) can be used as example. In this class, its attributes are mapped to attributes of the UML class *MultiplicityElement*, while the class *DMAssociationEnd* is mapped to the UML class *Property*. One might think that this situation is wrong because the semantics of the *DMAssociationEnd* is distributed between two UML classes. However, this situation is correct because the class *MultiplicityElement* is a generalization of the class *Property*. In other words, the class *Property* is a *MultiplicityElement*.

### 4.2 Systematic Approach to Define the Integration Metamodel

The systematic approach designed to facilitate the Integration Metamodel generation is composed of the following four steps:

- The first step: *Define the DSML metamodel related to the DSML of the MDA approach*. This definition must be elaborated according to the EMOF modeling capabilities defined in the MOF (Meta Object Facility) specification [7]. EMOF (Essential MOF) corresponds to a subset of the complete MOF specification (CMOF) that provides the essential metamodeling constructs. By using EMOF, the resultant DSML metamodel properties do not have features that are not supported by UML profiles. An example of this is the property redefinition of CMOF (not defined in EMOF) that is not supported by UML profiles. Moreover, EMOF has a standardized XMI definition [9]. Thus, the profile generation can be automated by means of model transformation rules that are implemented over the XMI definition of the Integration Metamodel. The EMOF definition also allows the implementation of specific editors with tools such as Eclipse GMF [2].
- The second step: *Perform the mapping between the DSML metamodel and the UML2 Superstructure*. The mapping must take into account: Classes, properties (attributes and associations), enumerations, enumeration literals, and data types. The mapping must be performed between elements of the same type (classes with classes, attributes with attributes, etc.). The mapped elements are considered as *equivalent elements* in the Integration Metamodel. For instance, in Fig. 2-B, the class *DMAssociation* is equivalent to the UML class *Association*. The mapping information must be stored in the Integration Metamodel definition.
- The third step: *Validate the OCL rules with the UML Superstructure*. The OCL rules defined in the Integration Metamodel must be validated to assure that they do not conflict with the OCL rules defined in the UML2 Superstructure. The validation is performed taking into account the OCL rules defined in the classes of the

Integration Metamodel and the corresponding classes of UML2 Superstructure (according to the equivalences identified in the previous step). This validation can be automated since the OCL rules can be computationally interpreted.

- The fourth step: *Fix mapping problems.* Identify the elements whose mappings violate the constraints defined in section 4.1 and divide its structure in order to fix them. The division must consider the structure and the semantics of the elements in the DSML metamodel and the mapped elements in the UML2 Superstructure. Then, for the new elements that are defined and the elements that are modified, repeat the all steps starting from the second one.

The systematic approach presented is iterative and finishes when all the mapping problems are solved. Fig. 3 shows how to apply the fourth step of this approach.

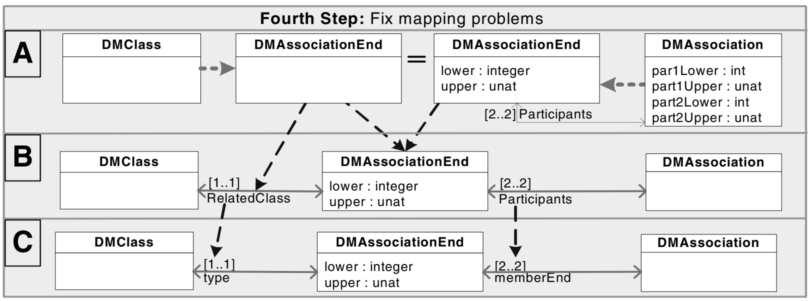



**Fig. 3.** Example of the fourth step for the systematic Integration Metamodel definition

Fig. 3 shows how the mapping problems are solved for the classes *DMClass* and *DMAssociation* presented in Fig. 2-A. As the figure shows, the semantics of the UML class *Property* (which represents the association end semantics) is located in both classes. To solve this situation, the structure related to this semantics is extracted from each class, defining a new class (Fig. 3-A). Since the extracted semantics represents the same conceptual constructs (association end), only one new class named *DMAssociationEnd* is defined. This new class has all the properties that are related to the association end semantics, and also maintains the relation with the class *DMClass* by means of the association *RelatedClass* (Fig. 3-B). Finally, to be consistent with the UML2 Superstructure, the associations *Participants* and *RelatedClass* are renamed to *memberEnd* and *type* (Fig. 3-C). Note that the association *RelatedClass* is equivalent to the UML association *type* because the UML class *Class*, which is mapped to the class *DMClass*, is a specialization of the UML class *Type*.

The second step of the proposed approach must again be applied over the new class obtained (*DMAssociationEnd*) as well as the modified classes (*DMClass* and *DMAssciation*). Finally, the Integration Metamodel is obtained (Fig. 2-B).

### 4.3 Implementing the Integration Metamodel

The implementation of the integration metamodel must be performed with a tool that supports the XMI standard, such as the Eclipse UML2 Tool [2]. By using this tool,

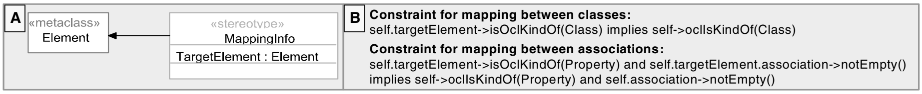



**Fig. 4.** Stereotype to store the mapping information of the Integration Metamodel

the defined model can also be exported as Ecore [2]. In order to store the mapping information within the Integration Metamodel definition, a stereotype over the UML class *Element* can be used (Fig. 4-A). The proposed stereotype can be constrained to validate that the mapping is performed between elements of the same type. Two examples of possible constraints for this stereotype are presented in Fig. 4-B

In editors in which only the EMOF definition is supported (like Eclipse EMF [2]) an annotation can be used in place of the proposed stereotype.

A good implementation practice is to document the Integration Metamodel as the UML2 Superstructure proposes. This practice helps to identify equivalences with UML and to understand the metamodel defined. The documentation should include: class name, brief description, semantics, generalizations, attributes, associations, OCL constraints, notation, generalization tree, guidelines, and some examples.

Even though we recommend storing the mapping information together with the Integration Metamodel definition, this is not mandatory. Other profile generation proposals, like the work presented in [14], might be interested in storing the mapping information separately. The main reason for the integration of the mapping definition is because this mapping information is not useful outside of the Integration Metamodel context. Therefore, following the systematic approach and the proposed implementation schema, all the needed information to generate a UML Profile is integrated in a unique XMI file defined according to the OMG Standards.

## 5 Benefits of the Integration Metamodel

The benefits related to using an Integration Metamodel to generate UML profiles are:

- *The definition of an Integration Metamodel is more intuitive than the definition of a UML Profile directly over the UML2 Superstructure.* This is because it is not mandatory to understand the concepts involved in a UML profile definition; for instance, how to define stereotypes or which UML metaclasses they extend.
- *The definition of an Integration Metamodel helps to isolate the complexity related to a UML profile design decisions.* All these design decisions are defined in the transformation rules involved in the automated profile generation. Thus, the effort to introduce changes in the UML profile (for instance, as a result of improvements performed to an MDA approach) is considerably reduced.

To exemplify the benefits mentioned, a transformation oriented to obtaining the stereotypes related to a specific modeling situation is analyzed (Fig. 5).

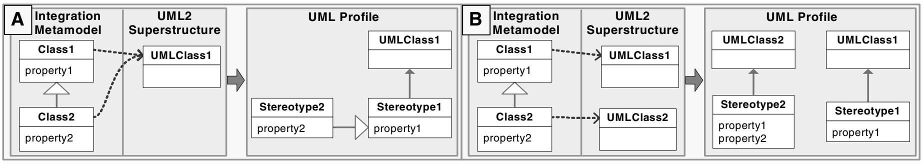



**Fig. 5.** Impact of a change in the Integration Metamodel for the generated UML Profile

Fig. 5-A shows a subset of a generic Integration Metamodel and the correspondent UML profile, the stereotypes *Stereotype1*and *Stereotype2* represent the semantics for the classes *Class1* and *Class2*, respectively. In Fig. 5-B, the mapping defined for the Integration Metamodel has been changed. This change generates a different profile according to the following transformation rule:

*"If there is a new generalization relationship between two classes from the Integration Metamodel that are equivalent to different classes of the UML2 Superstructure, the generalization relationship is not represented in the profile, the extensions of each stereotype to the correspondent UML class are defined, and the inherited non-equivalent properties are duplicated (Fig. 5-B)"*

A single change is introduced in the Integration Metamodel: the mapping of the class *Class2* is changed to a different UML class (Fig. 5-B). However, this change produces four changes in the resultant profile: 1) the generalization relationship is dropped, 2) the UML class *UMLClass2* is imported, 3) the extension for *Stereotype2* is defined, and 4) *property1* is added to the *Stereotype2*.

This basic example shows that it is more intuitive to change a mapping between classes than to define new extensions over the UML2 Superstructure. Also, the effort to obtain a new profile definition once the change has been performed is considerably reduced (one change in the Integration metamodel vs. four changes in the UML profile). Furthermore, since the complexity involved in the design of the UML profile is encapsulated in the transformation rule, it simplifies the work of the designer. In an Integration Metamodel designed for an industrial MDA solution, the impact of these changes can be even greater. This justifies the additional effort necessary to define the Integration Metamodel in order to automate the UML profile generation.

## 6 Conclusions and Further Work

In this work, an Integration Metamodel that improves the automatic generation of a UML profile has been introduced. The main purpose of this metamodel is to integrate the semantics of a DSML (related to an industrial MDA approach) into UML.

Taking into account the systematic approach presented and the benefits provided by the Integration Metamodel, the two specific requirements presented in the introduction of this paper have been satisfied. This is because the Integration Metamodel reduces the effort of introducing changes in the UML profile and facilitates the adaptation of the UML profile to the evolution of an MDA approach. In addition, this systematic approach allows the generation of the Integration Metamodel from the DSML metamodel of the MDA approach.

Apart from to the benefits mentioned, the Integration Metamodel can also be used as a mechanism to share knowledge between different MDA approaches because the structure and the mappings of the Integration Metamodel are aligned with the UML2 Superstructure. Therefore, a better understanding of the semantics and design decisions involved in different MDA approaches can be achieved taking the UML2 Superstructure semantics as reference. It is important to note that the concepts and ideas presented in this work can also be applied to improve MDA approaches that are already using UML and profiles as the modeling mechanism.

In order to validate this work, the Integration Metamodel for the OO-Method MDA approach [11] is currently being developed. The OO-Method approach has been selected because it has been successfully applied to the software industry through its model compiler [10]. Further work includes finishing the Integration Metamodel for the OO-Method approach and generating the equivalent UML profile following the proposed process. A tool for automatically generating the UML profile from the Integration Metamodel definition must also be implemented.

## References

1. Bruck, J., Hussey, K.: Customizing UML: Which Technique is Right for You? IBM (2007)
2. Eclipse: Eclipse Modeling Project, `http://www.eclipse.org/`modeling
3. France, R.B., Ghosh, S., Dinh-Trong, T., Solberg, A.: Model-driven development using uml 2.0: Promises and pitfalls. IEEE Computer 39(2), 59–66 (2006)
4. Fuentes-Fernández, L., Vallecillo, A.: An Introduction to UML Profiles. The European Journal for the Informatics Professional (UPGRADE) 5(2), 5–13 (2004)
5. Lagarde, F., Espinoza, H., Terrier, F., Gérard, S.: Improving UML Profile Design Practices by Leveraging Conceptual Domain Models. In: Proceedings of the 22th IEEE/ACM International Conference on Automated Software Engineering (ASE), pp. 445–448 (2007)
6. OMG: MDA Guide Version 1.0.1
7. OMG: MOF 2.0 Core Specification
8. OMG: UML 2.1.2 Superstructure Specification
9. OMG: XMI 2.1.1 Specification
10. Pastor, O., Molina, J.C.: Model-Driven Architecture in Practice: A Software Production Environment Based on Conceptual Modeling. Springer, Heidelberg (2007)
11. Pastor, O., Gómez, J., Insfrán, E., Pelechano, V.: The OO-Method Approach for Information Systems Modelling: From Object-Oriented Conceptual Modeling to Automated Programming. In: Information Systems, vol. 26(7), pp. 507–534. Elsevier Science, Amsterdam (2001)
12. Selic, B.: A Systematic Approach to Domain-Specific Language Design Using UML. In: Proceedings of the 10th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC), pp. 2–9 (2007)
13. Staron, M., Wohlin, C.: An Industrial Case Study on the Choice between Language Customization Mechanisms. In: Münch, J., Vierimaa, M. (eds.) PROFES 2006. LNCS, vol. 4034, pp. 177–191. Springer, Heidelberg (2006)
14. Wimmer, M., Schauerhuber, A., Strommer, M., Schwinger, W., Kappel, G.: A Semi-automatic Approach for Bridging DSLs with UML. In: Proceedings of the 7th OOPSLA Workshop on Domain-Specific Modeling (DSM), pp. 97–104 (2007)

# A UML Profile for Modelling Measurable Requirements*

Jesús Pardillo[1], Fernando Molina[2],
Cristina Cachero[1], and Ambrosio Toval[2]

[1] Lucentia Research Group,
Department of Software and Computing Systems,
University of Alicante, Spain
{jesuspv,ccachero}@dlsi.ua.es
[2] Department of Informatics and Systems,
University of Murcia, Spain
{fmolina,atoval}@um.es

**Abstract.** In the last years, the need for a sound integration of the *requirements engineering* discipline with the *model driven development* paradigm has promoted the definition of a myriad of requirements metamodels. However, most existing proposals still lack backward (alignment with business goals) or forward (connection with validation methods such as measures) traceability, which hampers their usefulness. In this paper, we present a 'measurable requirements metamodel' that connects goals, requirements, and measures, thus fostering a *goal-oriented measurable requirements engineering* perspective. In order to provide this metamodel with a familiar notation, we also present a UML profile based on the *i** framework, which facilitates its adoption in the context of any UML-based software engineering process.

**Keywords:** GORE, MDD, UML profile.

## 1 Introduction

The impact that requirements has on the success of software development projects has been assessed through a number of empirical findings [1,2] that reveal that some of the most important problems in software projects are due to their lack of alignment with business goals. With the aim of mitigating this situation, the *requirements engineering* (RE) discipline proposes a plethora of methods and techniques aimed at helping practitioners during the execution of requirements-related activities. Most of these methods (see *e.g.* [3]) have traditionally tackled the systematic elicitation of requirements using template-based requirement documents. However, in last years, *model driven development* (MDD) [4] proposals have introduced a new requirements management

* Partially supported by the Spanish projects ESPIA (TIN2007-67078), DEDALO (TIN2006-15175-C05-03), QUASIMODO (PAC08-0157-0668), and MELISA-GREIS (PAC08-0142-335). Jesús Pardillo and Fernando Molina are funded under the FPU grant AP2006-00332 and an FPI grant (Fundación Séneca), respectively.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 123–132, 2008.

perspective, based on requirements metamodelling. Also with the aim of reducing the potential problems associated with requirements, some RE approaches have worked on the explicit treatment of goals [5], thus coining the term *goal-oriented requirements engineering* (GORE). The MDD tendency has particularly affected GORE, with MDD-based GORE proposals such as [6,7], to name a few.

This notwithstanding, MDD-based GORE metamodels still lack support for important concepts, among which that of 'quantifiable requirement' outstands [8]. In this paper we claim that quantifiable requirements can be achieved through the integration of GORE metamodels with measurement metamodels into a 'measurable requirements metamodel' (MRM). This idea is developed in Section 2. The MRM metamodel comes together with a UML profile that provides the concrete syntax that serves to visually represent the metamodel concepts. Such profile is presented in Section 3 and it is based on an $i*$ framework profile, in an attempt to avoid the necessity for the RE community to learn 'yet another notation'. In order to further clarify the approach, Section 4 presents a case study that illustrates the application of the proposed modelling framework. Last, Section 5 gathers conclusions and further lines of research.

## 2 A Measurable Requirements Metamodel

As we have aforementioned, there are several GORE metamodeling proposals aimed at palliating the set of requirements-related problems detected in software projects. A recent study comparing existing proposals gave as a result a general requirements metamodel [9]. Although other, much more complete, metamodels could be used instead (see *e.g.* [10]), the proposal from [9] succeeds in being simple and yet providing us with the minimum set of concepts needed to stress the connection between goals and requirements. This metamodel is at the core of the requirements package of the MRM (from now on MRM-R). Such package is presented in Fig. 1.

However, this proposal on its own is not enough to deal with the 'quantifiable requirement' concept. Quantification can be achieved through the definition of measures that make a given requirement measurable [11]. The idea of defining measures together with requirements, far from being anew, is a practice that, included in many software development processes (see *e.g.* [12]), may significantly contribute to the project success. In order to define measures associated with requirements, we have used the well-known 'software measurement metamodel' (SMM) from García *et al.* [13]. Some adjustments were needed however: on one hand, some of the concepts in SMM were defined at different levels of abstraction, not suitable for our purpose. Others simply overlapped with the concepts gathered in the MRM-R. The result of these adjustments is the measurement metamodel package (MRM-M) depicted in Fig. 2.

Back to Figs. 1 & 2 we can observe how the linking point between the MRM-R and the MRM-M is (i) the `Validation Method` abstract metaclass on one hand and (ii) the `Indicator` metaclass (a type of measure) on the other.

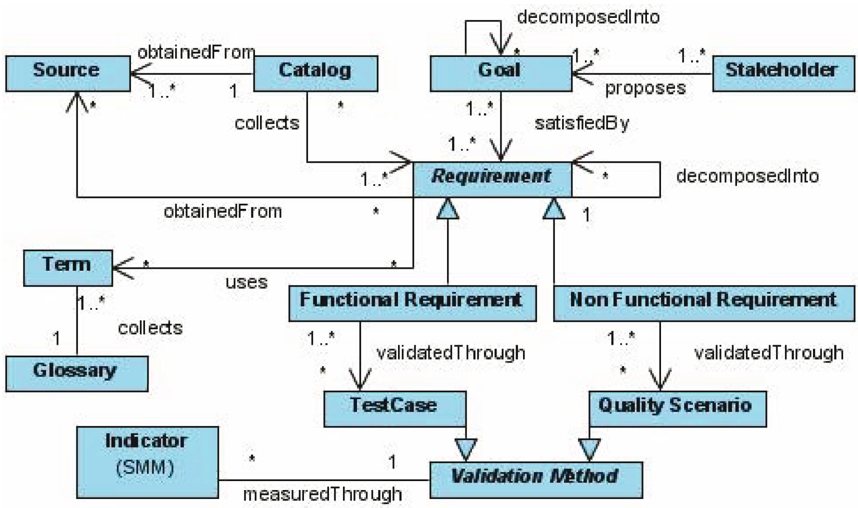



**Fig. 1.** Goal-oriented requirements metamodel package together with the MRM-M link

It is important to note how the integration of these two packages enriches both existing requirements metamodels, which now can define measurable requirements, and existing measurement metamodels, which now can soundly express the context and purpose of the measures. In MRM-R, this context not only involves specific goals or requirements, but it also includes specific quality scenarios with conditions under which the measures are to be applied. All this information is of paramount importance for the definition of measures, as they affect the acceptability of the measurement result. For example, applying a performance measure with a system load of 10 simultaneous users (a particular quality scenario) is different from applying the same measure with 5000 simultaneous users (a different one).

However, even with the aid of a soundly defined metamodel such as MRM, achieving models that are at the same time concise and complete is far from easy, and can become too cumbersome if we lack the adequate notation mechanisms [14]. For our proposal, we have evaluated two different notational approaches: (i) defining a synthesised concrete syntax (icons) or (ii) instantiating the metamodel as an object model in a heavy-weighted metamodelling approach. In our experience, object models tend to be rather cumbersome as the system increases its complexity [15]. This notwithstanding, a new iconography has the risk of forcing the modellers to learn 'yet another notation' if not chosen carefully.

Our solution to this problem has been defining a synthesised concrete syntax, and overcoming the 'learning overload' drawback by associating the MRM concepts with the iconic notation defined in $i*$, a well known RE framework. This notation reuse is based on semantic equivalencies. As an example, the MRM `Measure` concept can be visualised by means of an $i*$ `Goal` icon, because a measure is a property of the software that we want to achieve. This notation

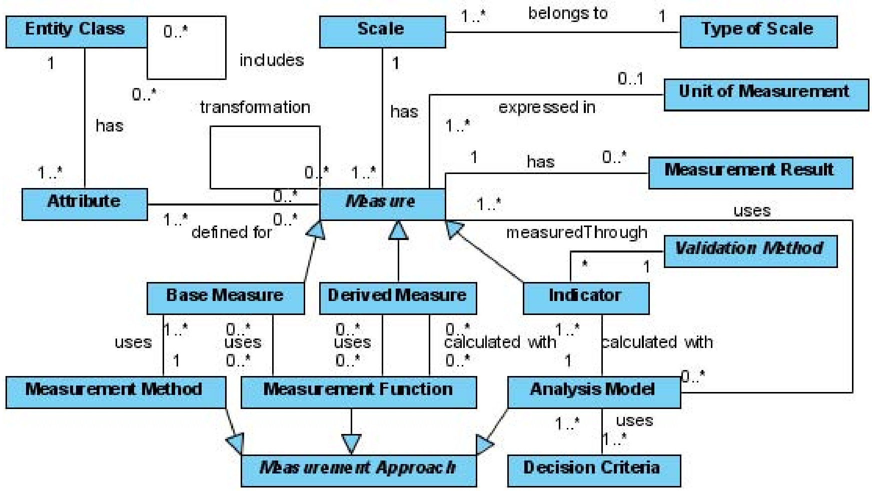



**Fig. 2.** Requirements-related measurement metamodel package together with the MRM-R link

reuse has been achieved through the definition of a UML profile that, based on a previously existing $i*$ profile [16], is presented next.

## 3 UML Profile for Measurable Requirements Modelling

The 'unified modelling language' (UML) [17] is a general purpose language, recognised as the standard notation for software artefacts specification [18] in MDD approaches. Nevertheless, sometimes it is useful to specialise it for particular domains or to enrich modelling perspectives. For this specialisation, UML profiles provide an easy-to-use way to extend UML. These profiles are articulated around stereotypes, tag definitions and constraints usually defined in the 'object constraint language' (OCL). By profiling, we can easily and powerfully implement domain-specific language notations that can be integrated into UML-based development processes and tools.

Given its usefulness, a UML profile has already been defined for $i*$ [16] (see Fig. 3, $i*$ profile). This profile defines the $i*$ concepts as UML stereotypes, and associates $i*$ icons to these stereotypes. However, in order for this UML profile to be able to deal with the kind of quantifiable requirements gathered in MRM, we need additional modelling constructs, not contemplated in the original $i*$ profile. Such new constructs are presented in Fig. 3, inside the MRM profile package.

In the $i*$ profile included in this figure we can observe how, given the `Class`, `Package`, `Association` and `AssociationClass` UML metaclasses, it is possible to extend them (by means of the UML *Extension* relationship) to support the different $i*$ concepts (goals, resources, means-end relationships, etc.) as stereotyped metaclasses.

Furthermore, this profile can be used as a basis to specialise the MRM concepts. This specialisation has been performed in two steps. First, we have directly reused all the *i** profile concepts that are included without modifications in MRM. That is the case of `Goal` or the different kinds of intentional relationships (`iRelationship` hierarchy). Second, for those concepts that were semantically similar (but not equal) to existing *i** stereotypes, we have defined new stereotype generalisation hierarchies. For example, back to Fig. 3, we can observe how the `Functional Requirement` concept (`FR`) has been defined in the MRM profile as a specialisation of the `Task` stereotype (imported from the *i** profile). The same occurs with `NFR`, `Stakeholder`, `Decision Criteria`, and so on.

However, not all the concepts included in MRM have such a straightforward connection with *i** stereotypes. Some of them (see *e.g.* `Requirement`, `Validation Method`, or `Measurement Approach`) are abstract metaclasses that, given the fact that they cannot be instantiated, do not require concrete syntax elements, and therefore they can be omitted from the profile. Other MRM metaclasses, namely the `Glossary` and `Term` metaclasses, have not been provided with a modelling mechanism, because their textual nature makes them poor candidates for graphical representation. Still other MRM elements, such as `Quality Scenario` or `Test Case`, have been defined as an specialisation of the

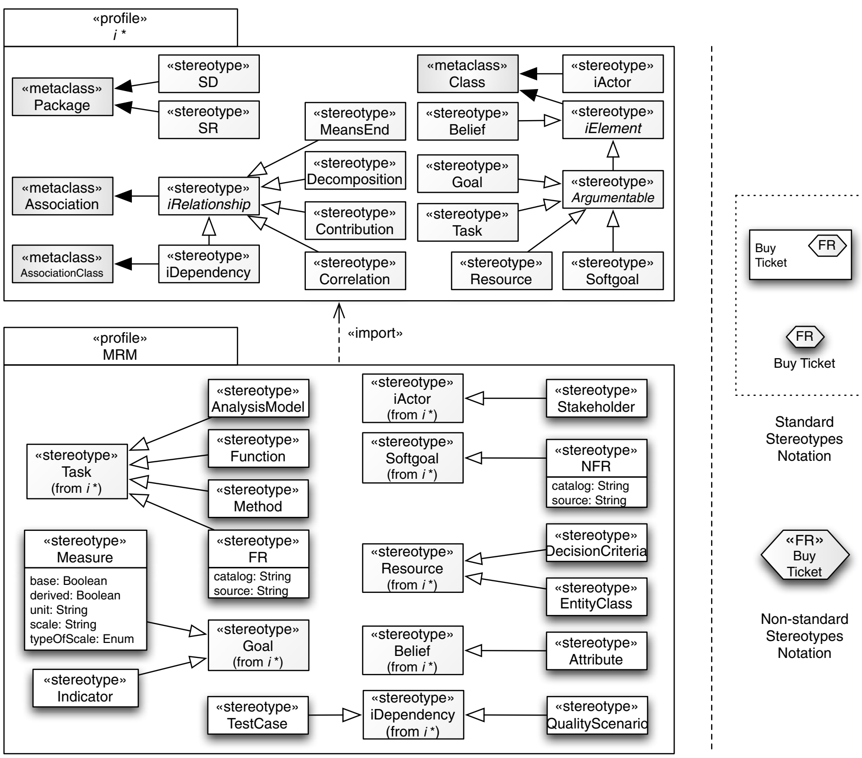



**Fig. 3.** MRM profile for measurable requirements modelling

$i^*$ dependency concept, as they connect the involved stakeholders. Last but not least, some concepts did not fit with the semantics of any $i^*$ profile construct. That is the case of `Source`, `Catalog`, `Unit of Measurement`, `Scale` and `Type of Scale`. For these elements, we have included a set of homonym tag definitions in the context of the related stereotypes (see Fig. 3). Due to space constraints, a deeper discussion of the rationale behind all these decisions has been left out of this paper.

We have also used a tag definition to distinguish between `Base Measure` and `Derived Measure`, both of which are represented in our profile with a `Measure` stereotype. The reason for doing so instead of defining two different stereotypes is that we do believe that the way a given measure is calculated is not relevant for a measurable requirement model, and therefore we have provided the same stereotype (and related notation) for both concepts. Generally speaking, an important advantage of tag definitions is that, given the fact that they do not require an explicit visualisation construct, its use generates models that are more concise.

The definition of the MRM profile as an specialisation of a general $i^*$ profile makes possible the automatic reuse of the $i^*$ notation, the only difference with respect to the original profile being the set of stereotypes that are printed on these icons (see Fig. 3). Regarding this notation, we have identified three alternatives that can be seen on the right-hand side of Fig. 3. For the sake of conciseness, from now on we will use the third option (see Fig. 4). Additionally, we would like to note how the stereotype names follow the measurement ontology defined in [13], thus further augmenting the understandability of the model for measurement experts.

As it occurs with almost any UML profile, we have defined several OCL constraints to preserve the accurate definition of MRM models. For instance, OCL is used to restrict the valid use of $i^*$ relationships according to the relationships among the MRM metaclasses. Due to space constraints, a further discussion of these OCL restrictions is left out of the scope of this paper.

We would like to finish this section by noting how, according to [19], the stereotypes introduced by the $i^*$ profile can be classified as *redefining* stereotypes, whereas the stereotypes of the MRM profile fall under the *restrictive* category. This means that, while the original $i^*$ extension redefined UML, the MRM extension is a means of representing MRM in the $i^*$ notation, thus truly reusing the $i^*$ concepts.

To further clarify our approach, Section 4 shows a case study that models a set of functional and non functional measurable requirements with the aid of the MRM profile.

## 4 Case Study

The case study chosen to illustrate our approach deals with a requirements elicitation scenario for an on-line cinema ticket sales system.

For the sake of simplicity, let's assume that the *sales manager* of a certain cinema chain wants to*increase the sales profit* of the company. Let's also suppose that this goal is to be achieved through two different sub-goals; on one hand the company wants to *increase the sales net profit* by implementing a web-based on-line ticket sale system that decreases the costs associated with the sales process. On the other hand, the company also wants to *increase the number of sales* by reaching a broader range of customers. The sales manager believes that offering the tickets through Internet may positively influence both sub-goals, so s/he has embarked into a web ticket sales system development process.

As for the *web customer* that is going to interact with the application, s/he has as the main functional requirement that of *buy ticket*. This requirement can be decomposed into two functional requirements: *browse films* and *purchase ticket*. In addition, several non-functional requirements have been identified. The *buy ticket* functionality should follow *accessibility* guidelines to allow web customers with disabilities to access the system. Additionally, the system should provide *information accuracy* while browsing through the films: synopsis, sessions, prices and so forth should be reliable. Also, the application *learnability* should be high, that is, the application should be simple enough for novel users to easily learn its operation. Last, the purchase process should be performed assuring the *security* of the customer data.

### 4.1 Instantiating the MRM Metamodel

If we first check the involved MRM-R metaclasses (see Fig. 1 & 4), we can observe how (i) the sales manager and web customer both instantiate the `Stakeholder` metaclass, (ii) all goals related to increasing sales profit instantiate the `Goal` metaclass, (iii) buy tickets, browse films, purchase tickets are all `Functional Requirement` occurrences and (iv) accessibility, learnability, security and information accuracy are all `Non Functional Requirement` instances.

Let's now focus on the learnability with which the web customer is able to browse the films we are offering on our site, and let's turn it into a measurable requirement. For this purpose, let's assume that we have defined a quality scenario (instantiated from the `Quality Scenario` metaclass, see Fig. 1 & 4) that states that it is necessary to check whether a novel user is able to find the link s/he expected based on his/her mental model of the domain.

If we now check the MRM-M (see Fig. 2 & 4), we can observe how the first step for making a requirement measurable is defining an `Indicator` occurrence that we have named 'navigability level'. Indicators are made up of measures. In our example, let's imagine there is a single `Derived Measure` instance that contributes to the indicator, which is called 'domain coverage navigation measure' (DCNM). This measure is based on two `Base Measure` occurrences: the 'number of relevant relationships' (NRR) and the 'number of navigated domain relationships' (NNDR). Briefly speaking, the DCNM calculates how many of the domain relationships that may be relevant for fulfilling the requirement are actually navigation links available through the application interface. For space reasons, readers interested in the whole rationale behind this measure as well as a whole SMM metamodel instantiation are referred to [15].

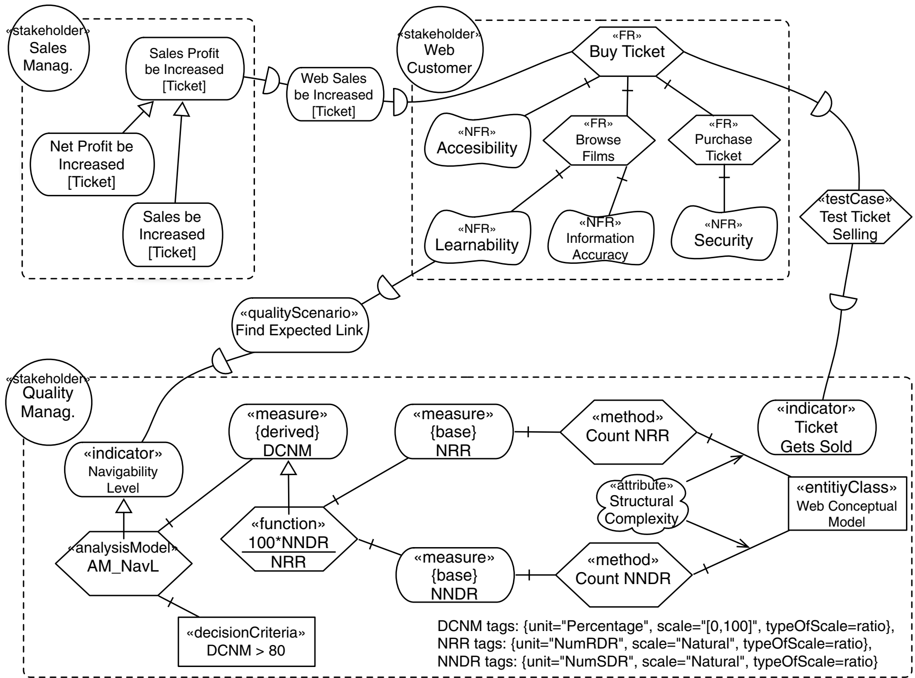



**Fig. 4.** MRM requirements model: the on-line cinema ticket sales system

Let's also imagine that, in this context, the `Decision Criteria` and the `Analysis Model` associated with the indicator establish that a *DCNM > 0.80* makes the web design acceptable with respect to its learnability. Assuming that the system is being deployed with a web engineering proposal such as OO-H [20], this measure can be calculated early in the lifecycle, based on the `Entity Class` 'web conceptual model' (an abstract model that represents all the views involved in a web application). Both measures are related with the structural complexity `Attribute` of the models associated. The `Measurement Function` that serves to calculate the DCNM uses the two `Base Measure`s in order to calculate the measurement result. The higher this value is, the more domain relationships are covered in the application interface by means of links.

### 4.2 Visual Modelling by Reusing the *i** Notation

As we have aforementioned, the MRM profile provides an easy way to visually specify MRM models. These models are not only intuitive for software analysts familiar with *i**, but also understandable by any UML modelling tool. The MRM model corresponding to our case study is shown in Fig. 4. As the reader may notice, the icons and stereotypes of the MRM profile give the MRM models an appearance that is very similar to that of traditional *i** diagrams. For example the 'buy ticket' functional requirement is represented by means of the ⬭ icon labelled with the «FR» stereotype (Fig. 4). Other examples are the representation of the 'navigability level' by means of a goal icon (⬭) labelled as «indicator» or the «entityClass» resource (▭) 'web conceptual model' that is later decomposed into the corresponding web-related models.

*Implementation.* The discussed case study and the entire modelling architecture that we propose in this paper have been implemented in the *Eclipse* development platform, which hosts a number of MDD-related projects that have allowed us to easily define the MRM profile and the associated tool palettes.

## 5 Conclusions

In this paper, we have presented a modelling solution based on UML profiles for representing measurable requirements. This solution consists of:

- a merging process between a GORE requirements metamodel (MRM-R) and an ontology-based measurement metamodel (MRM-M) into a measurement requirements metamodel (MRM). MRM merges goals, requirements and measures, providing backward and forward traceability, and it paves the way for the inclusion of measurable requirements in MDD environments.
- an MRM profile that allows the integration of our proposal into UML-based development approaches and tools and helps to overcome some of the well-known UML deficiencies for specifying complex environments and requirements models [21]. This profile, far from being 'yet another profile', consists of a light-weighted extension to a previous $i$* profile [16]. In this way we have reused the $i$* concrete syntax (notation), therefore avoiding, again, the need to learn 'yet another notation'.

Some future lines of research include (i) the extension of MRM-R to deal with complex requirements topics such as conflicting requirements and (ii) an empirical evaluation of the suitability of MRM for requirements elicitation in practice. We also plan to align our modelling solution with a model-transformation architecture, as the MDD philosophy suggests, in order to automatically derive tool-usable metadata from MRM models.

## References

1. Glass, R.: Software Engineering: Facts and Fallacies. Addison-Wesley, Reading (2002)
2. Group, T.: The CHAOS Report (2002), http://www.standishgroup.com/
3. Toval, A., Nicolas, J., Moros, B., García, F.: Requirements Reuse for Improving Information System Security: A practicioner's Approach. Requirements Engineering Journal 6(4), 205–219 (2002)
4. Bézivin, J.: Model Driven Engineering: An Emerging Technical Space. In: GTTSE, pp. 36–64 (2006)

5. Yu, E.S.K.: Towards Modeling and Reasoning Support for Early-Phase Requirements Engineering. In: RE, pp. 226–235 (1997)
6. Berre, A.J.: COMET (Component and Model based Development Methodology) (2006), http://modelbased.net/comet/
7. Bolchini, D., Paolini, P.: Goal-driven requirements analysis for hypermedia-intensive Web applications. Requir. Eng. 9(2), 85–103 (2004)
8. Blaine, J.D., Cleland-Huang, J.: Software Quality Requirements: How to Balance Competing Priorities. IEEE Software, 22–24 (2008)
9. Molina, F., Pardillo, J., Toval, A.: Modelling Web-based Systems Requirements using WRM. In: WISE Workshops (in Press, 2008)
10. Franch, X., Grau, G., Quer, C.: A Framework for the Definition of Metrics for Actor-Dependency Models. In: RE, pp. 348–349 (2004)
11. Glinz, M.: A Risk-Based, Value-Oriented Approach to Quality Requirements. IEEE Software, 34–41 (2008)
12. Larman, C.: Applying UML and Patterns (2005)
13. García, F., Bertoa, M.F., Calero, C., Vallecillo, A., Ruiz, F., Piattini, M., Genero, M.: Towards a consistent terminology for software measurement. Information & Software Technology 48(8), 631–644 (2006)
14. Ernst, N., Yu, Y., Mylopoulos, J.: Visualizing non-functional requirements. In: REV, vol. 6, pp. 2–12 (2006)
15. Cachero, C., Meliá, S., Genero, M., Poels, G., Calero, C.: Towards improving the navigability of Web applications: a model-driven approach. European Journal of Information Systems 16, 420–447 (2007)
16. Mazón, J.N., Pardillo, J., Trujillo, J.: A Model-Driven Goal-Oriented Requirement Engineering Approach for Data Warehouses. In: ER Workshops, pp. 255–264 (2007)
17. Object Management Group: Unified Modeling Language (UML), version 2.1.1 (February 2007), http://www.omg.org/technology/documents/formal/uml.htm
18. Bézivin, J., Muller, P.A.: UML: The Birth and Rise of a Standard Modeling Notation. In: Bézivin, J., Muller, P.-A. (eds.) UML 1998. LNCS, vol. 1618, pp. 1–8. Springer, Heidelberg (1999)
19. Berner, S., Glinz, M., Joos, S.: A Classification of Stereotypes for Object-Oriented Modeling Languages. In: UML, pp. 249–264 (1999)
20. Gómez, J., Cachero, C., Pastor, O.: Conceptual Modeling of Device-Independent Web Applications. IEEE MultiMedia 8(2), 26–39 (2001)
21. Glinz, M.: Problems and Deficiencies of UML as a Requirements Specification Language. In: IWSSD, pp. 11–22 (2000)

# A Comprehensive Aspect-Oriented Use Case Method for Modeling Complex Business Requirements

Caimei Lu and Il-Yeol Song

College of Information Science and Technology, Drexel University
Philadelphia, PA 19104, USA
{Caimei.lu,song}@drexel.edu

**Abstract.** The aspect-oriented approach separates cross-cutting concerns and models them as aspects. In this paper, we present a comprehensive aspect-oriented use case approach for modeling complex business requirements. We identify four categories of aspects: high level non-functional requirements, extending or optional requirements, included or subordinate requirements, and business rules. Our aspect-oriented use-case model comprises three different templates: base use cases, aspectual use cases, and nonfunctional aspects. The proposed method offers a systematic approach for identifying different potential aspects from complex business requirements. Moreover, our method provides a comprehensive model for documenting identified aspects and their composition to the core functionalities at an early stage of software development. The contributions of our paper include identifying and utilizing the four types of aspects and proposing the templates to document them.

**Keywords:** Early Aspects, Aspect-oriented modeling, Aspect-oriented system development, aspect-oriented use case method, Aspect template.

## 1 Introduction

The aspect-oriented paradigm has been proposed to address the cross-cutting concerns or requirements that cannot be cleanly encapsulated in a single class. The notion of "Aspect" was first proposed in the context of Aspect-Oriented Programming (AOP) [4], which refers to the "property for which the implementation cannot be cleanly encapsulated in a generalized procedure". Recently, various aspect-oriented analysis and design approaches have been proposed to identifying, modeling, and representing "aspects" at earlier stages of software development.

This paper focuses on how to identify and model aspects at the requirement engineering stage. We extend the use case approach with aspects to address both functional and non-functional requirements. Here, aspects are defined as the requirements cross-cutting multiple use cases. Addressing cross-cutting concerns at the use case level allows us to detect and resolve conflicts at an early stage. Thereby, the code-level conflicts caused by cross-cutting concerns could be addressed much earlier before the implementation stage.

In this paper, we propose *four categories of aspects* existing in complex business requirements: *non-functional requirements* (NFRs), *extending or optional requirements*,

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 133–143, 2008.

*included or subordinate requirements*, and *business rules*. We first propose a comprehensive aspect-oriented use case method that models all these four categories of aspects from the core applications at the early stage of system development. We then present how to document our approach using three different types of templates.

Many approaches have been proposed for modeling aspects and composing them with the base model at the requirements level. They include the scenario-based approach, the goal-oriented approach, the component-based approach, the multidimensional concern modeling approach, and the theme/doc approach. Our paper belongs to the approach that adapts use cases for modeling aspects. Araújo and Rashid et al. attempt to incorporate aspects into UML elements [1, 2, 5]. The authors propose a general model for aspect-oriented requirements engineering, which includes three main parts: identifying cross-cutting concerns, specifying functional concerns, and composing requirements. In [2], Araújo et al. refine the process model to combine viewpoints, use cases, and aspects. Here, the aspects are identified from two sources: (1) NFRs that cross-cut a number of use cases and (2) aspectual use cases—use cases that extend or are included by more than one use case.

Jacobson and Ng [3] propose an approach called the aspect-oriented software development with use cases. This approach identifies and represents aspects with use cases. They view uses cases as cross-cutting concerns in that the realization of each use case cut across many classes. Accordingly, they point out that “when you identify use cases, you are indeed identifying aspects.” In our paper, however, we adopt the definition of aspects used by Araujo [1, 2, 5] in that *a requirement is viewed as an aspect only when it cross-cuts more than one base use case*.

Another feature that differentiates the use case model proposed by Jacobson and Ng from the traditional use case model is that NFRs are separately represented with so-called infrastructure use cases. An infrastructure use case is a construct dealing with system activities needed to meet system-wide infrastructure concerns.

Our model is based on Jacobson and Ng’s research [3] and Araujo et al.’s research [1, 2]. Following Jacobson and Ng, we introduce AOP concepts such as “join points” and “pointcuts” used to compose aspects into base use cases. Following Araujo et al.’s method, we specify that NFRs, extending requirements, or included requirements are viewed as aspects only when they cross-cut more than one base use case.

However, the use case model proposed here is different from either of theirs. Our method is different from Jacobson and Ng’s method in the following three points:

- Our use case model is based on a different definition of aspects at the requirements level. While Jacobson and Ng employ the three types of aspects: NFRs, peers, and extensions, we adopt the four types aspects: NFRs, extending requirements, included requirements, and business rules. A *peer* is an independent requirement that can be used separately with no reference to each other. Thus, we do not consider “peers” as aspects, because peer use cases are independent of each other *at the requirements level*. The entanglement between peers only appears until the design stage when the class diagram is used for realizing different peer use cases.
- While Jacobson and Ng represent high level NFRs using low-level infrastructure use cases, we represent them with a special NFR template. We consider it is improper to identify these low-level infrastructure use cases at the requirement level, because behaviors described by these use cases are invisible to the user, and there

is no action involved from the user's side in these infrastructure use cases. Therefore, we use a high-level template to describe the nonfunctional aspects.

- Jacobson and Ng treat all extensions as aspects, even if an extension is related with only one base use case. This view violates our definition of the aspect. In this paper, we treat an extending requirement as an aspect only when it cross-cuts more than one base use case.

Our aspect-oriented use case method extends the method by Araujo et al. in the following three points:

- They consider only three types of cross-cutting aspects: NFRs, extending requirements, and included requirements. We also consider cross-cutting business rules as the 4$^{th}$ kind of aspects.
- We proposed new and specific templates for modeling aspectual use cases.
- We define *join points* in the base use cases and *pointcuts* in aspectual use cases to facilitate the composition between base use cases and aspectual use cases.

Thus, the contributions of our paper include classifying aspects into four categories, proposing the templates to model them, and proposing a method of composing aspects into base use cases.

The rest of this paper is organized as follows: Section 2 introduces the integrated use case method for modeling the four types of aspects. Section 3 provides a case study to illustrate the proposed method. Finally, we conclude our work in Section 4.

## 2 The Framework of Aspect-Oriented Use Case Modeling

In this section, we introduce the framework of our aspect-oriented use case modeling method. Fig. 1 shows the process model of the aspect-oriented use case method.

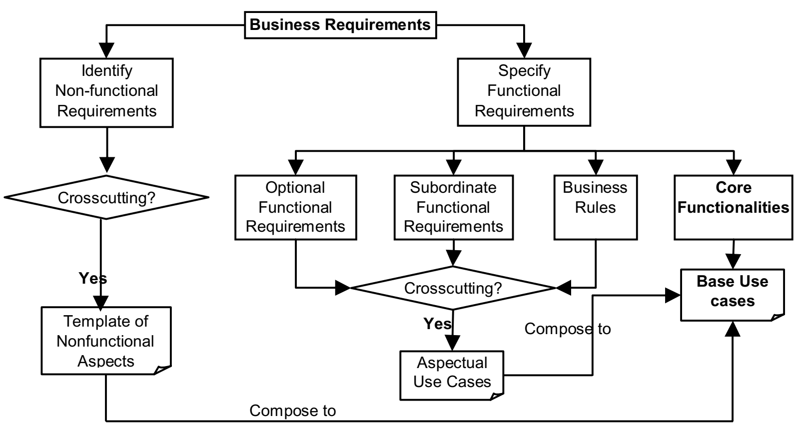



**Fig. 1.** The Process Model of an Aspect-oriented Use Case Method

## 2.1 Specify Functional Requirements

Most expressed business requirements are functional requirements which define what the system should do. From the functional requirements we can further identify the core functionalities which define the primary flow of the system process. Besides the core functionality, we can identify other three categories of functional requirements which are the sources of aspects. The first category is *optional requirements*, which describe the alternative or extending flows of the primary flow under a special condition. An example is to create a back order when the ordered item is not available. The second category is *subordinate functional requirements*. Subordinate functional requirements are usually represented with inclusion use cases. For example, p*rocess credit card* may be an included use case of a base use case called *place order*. The third category is *business rules*. Business rules are very changeable, so it is beneficial to model them separately without invasively influencing the core applications. Business rules are usually domain-specific. There are some common models of business rules across different applications, such as price personalization. If these common models can be represented with aspects, they could be plugged in and out of the application and reused across different applications.

## 2.2 Identify Non-functional Requirements

Complex business requirements include not only requirements directly related to the functionality of the system, but also some NFRs which constrain the functionality of the system. NFRs are usually system-wide quality concerns described as declarative statement such as performance, security, response time, reliability, etc. NFRs are widely-accepted aspects.

## 2.3 Build Base Use Cases

Base use cases model the core functionalities of the system. *The template of our base use cases is shown in Fig. 1*. It is different from typical UML use case template in three respects. First, the proposed template incorporates an additional component called "*Join points*" to identify the points where aspectual use cases are composed to the base use case (see Table 1). The concept of the *join point* is from AspectJ, in which it indicates an execution point where additional behaviors will run [1]. Each *join point* has a unique label which will be refereed by aspectual use cases. The "*Position*" states *before* or *after* or *within* which specific step of the primary flow the join point is located.

Second, the proposed template of a base use case defines a component called "Non-functional Aspects", which list all the cross-cutting NFRs that affect the current base use case. Third, the "Special Requirements" component defined in the proposed template merely includes NFRs or business rules which are not cross-cutting and only affect current base use case. The proposed template of base use case also include "Extension Use Cases" and "Inclusion Use Cases" components, which list use cases modeling non-cross-cutting optional requirements and subordinate requirements associated

**Table 1.** The Template for the Base Use Case

| | | |
|---|---|---|
| ***USE CASE Name*** | | |
| ***ACTOR*** | | |
| ***Goal*** | | |
| ***Level*** | **Base** | |
| ***Trigger*** | << What event starts the use Case>> | |
| ***The Primary Flow in numbered sequence*** | **Actor Action** | **System Action** |
| | | |
| ***Extension Use Cases*** | | |
| ***Inclusion Use Cases*** | | |
| ***Special Requirements*** | | |
| ***Nonfunctional Aspects*** | <<List of NF Aspects that affect this use case>> | |
| ***Join Points*** | **Label** | **Position** |
| | JP1 | |
| | JP2 **…** | |

with current base use case. Note that when a base use cases are initially built, the components of "Nonfunctional Aspects" and "Join Points" are left unfilled, because we can not tell what NFRs, or functional aspects are connected to the base use case until the nonfunctional aspect template and aspectual use cases are built. So these two important components are filled while the nonfunctional aspect templates and aspectual use cases are developed.

## 2.4 Identify and Create Functional Aspectual Use Cases

After all optional functional requirements, subordinate functional requirements, and business rules are analyzed from the business requirements, their relationships to the base use cases are examined. Those requirements conforming to any of the following three conditions are modeled as *aspectual use cases*:

- Optional requirements which extend more than one base use case
- Subordinate requirements which are included by more than one use case
- Business rules which affect more than one use case

Those requirements that do not conform to the above three conditions are modeled in the traditional way: optional requirements are modeled with the extension use cases; subordinate requirements are modeled with inclusion use cases; business rules are modeled as special requirements of the corresponding base use case.

Table 2 shows *the template of an aspectual use case*. The level of the use case is *Aspect*, and the type of the use case is either extension, inclusion, or business rule. In correspondence to the *join points* defined in base use cases, we define "Pointcut" and "Advice" in aspectual use cases. Each *pointcut* corresponds to a set of *join points* (at least two since each aspectual use case is composed to more than one base use case). Each *pointcut* has a label as its unique identifier. Each corresponding *join point* is referred through the combination of the name of the base use case, in which the *join point* is defined, and the label of the *join point*. *Advice* describes the additional flow composed to the base use case at each *join point* of the *pointcut*.

### 2.5 Identify and Model Non-functional Aspects

NFRs always deal with system-wide quality concerns such as performance and reliability and always work in the background of the system. Therefore, they cannot be modeled directly with use cases. Following the method of [1, 2, 6], we use a special template to describe the nonfunctional aspects, as shown in Table 3. However, not all NFRs are modeled with the nonfunctional aspect template. After identifying all the NFRs, we first check their associations with the base use cases. If a nonfunctional requirement affects more than one use case, it is modeled with the non-functional aspect template; otherwise, it is only listed as a special requirement in the base use case it affects. In Table 3, the "*Influence*" line lists all the base use cases affected or cross-cut by the current non-functional aspect. The relations of the current non-functional aspect to other non-functional aspects are also specified to help designers understand the constraints and trade-off among different cross-cutting NFRs.

**Table 2.** Template for Aspectual Use Case

<table>
<tr><td>USE CASE Name</td><td colspan="2"></td></tr>
<tr><td>ACTOR</td><td colspan="2"></td></tr>
<tr><td>Goal</td><td colspan="2"></td></tr>
<tr><td>Level</td><td colspan="2">Aspect</td></tr>
<tr><td>Type</td><td colspan="2">( Extension | Inclusion | Business Rule )</td></tr>
<tr><td>Condition</td><td colspan="2"></td></tr>
<tr><td rowspan="2">Advice</td><td>Actor</td><td>System Action</td></tr>
<tr><td></td><td></td></tr>
<tr><td>Pointcut</td><td>Label</td><td>Corresponding Join Points</td></tr>
<tr><td></td><td></td><td></td></tr>
</table>

**Table 3.** Template for Non-functional Aspect

<table>
<tr><td>NFR Name</td><td colspan="2"></td></tr>
<tr><td>Goal</td><td colspan="2"></td></tr>
<tr><td>Overview</td><td colspan="2"></td></tr>
<tr><td>Influence</td><td colspan="2"><< Use cases influenced by this Aspect >></td></tr>
<tr><td rowspan="2">Relations to other Non-functional Aspects</td><td>Positive</td><td>Negative</td></tr>
<tr><td><<Other NF Aspects positively related to this Aspect>></td><td><<Other NF Aspects negatively related to this Aspect>></td></tr>
<tr><td>Priority</td><td colspan="2"><< Classify into first, second, third, depending on the importance>></td></tr>
</table>

## 3 A Case Study

In this section, we use a case study of Subscription Automation System (SAS) to illustrate the Aspect-oriented use case method presented in Section 2. The requirements of SAS are stated as follows:

The SAS is built for a monthly magazine called HOST (Hot Software Technologies) published by a company. HOST is sold on a subscription basis. The subscription process is Web-enabled. Most subscriptions are for a one year period. If individuals subscribe for multiple-years, they receive a 10% discount for each year up to three years. Subscriptions can come by mail, phone, or the Web. The Web subscription requires the use of a credit card for payment, while other subscription could be paid for by a check or a credit card. When subscribing through Web, one can also check the subscription status or renew. The termination of a subscription, however, cannot be done via Web. The termination must be initiated by the customer via a phone, fax, or a written request and must be processed by a staff member. For each subscription and cancellation, a confirmation notice is sent via email to the subscriber. Some subscribers live in foreign countries and issues need to be mailed to the foreign countries with a proper mailing charge. The current foreign postal charge is $25 per year. The company also offers free subscriptions. If the subscriber is the author of articles or the member of the editorial board, the subscription to the HOST is free. When the customer subscribes through the Web, the response time for all checking and displays should be within 5 seconds.

### 3.1 Specify Functional Requirements

First, we identify the core functional requirements of the system. We then specify the optional requirements, subordinate requirements, and business rules contained in the functional requirements. The core functionality of the system is represented in the use case diagram of Fig. 2(a). Other functional requirements are listed as follows:

- *Optional requirements:*
  - **OR1:** If the subscriber is the author of articles or the member of the editorial board, the subscription to the HOST is free.
- *Subordinate Requirements:*
  - **SR1:** The web subscription requires the use of a credit card for payment, while other subscription could be paid for by a check or credit card.
  - **SR2:** For each subscription and cancellation, a confirmation notice is sent via email to the subscriber.
- *Business Rules:*
  - **BR1:** If individuals subscribe for multiple-years, they receive a 10% discount for each year up to three years.
  - **BR2:** Subscribers in foreign countries are charged extra $25 for the postal fee.

### 3.2 Identify Non-functional Requirements

The only NFR identified in this case is **Response Time:** *response time for all checking and displays should be within 5 seconds.*

### 3.3 Build Base Use Cases

According to the use case diagram presented in Fig. 2(a), we have eight base use cases which represent the core functionality of the system. Here we only present the use case of "Add subscription" as an example (shown in Fig. 3(a)). Note that the five *join points* and nonfunctional aspects are added later after Step 3.4 and Step 3.5.

### 3.4 Identify and Create Functional Aspectual Use Cases

After all the base use cases are built, we examine if each optional requirement, subordinate requirement, and business rule cross-cuts more than one base use case. If so, it is modeled with an aspectual use case. At the same time, all the corresponding *join* *points* of its *pointcut* are identified. The cross-cutting relationship can be examined based on the use case diagram. From the use case diagram presented in Fig. 2(b), we can see that:

- **OR1** cross-cuts two base use cases “Add Subscription” and “Renew/Change subscription”;
- **SR1** cross-cuts three use cases “Add Subscription”, “Renew/Change subscription,” and “Cancel Subscription”;
- **SR2** cross-cuts four use cases “Maintain Subscriber”, “Add Subscription”, “Renew/Change subscription,” and “Cancel Subscription”;
- **BR1** cross-cuts two use cases “Add Subscription” and “Renew/Change subscription”;
- **BR2** cross-cuts two use case “Add subscription” and “Renew/Change subscription”.

Consequently, we build an aspectual use case for each of the above requirements. Here we only present the aspectual use cases for OR1 (shown in Fig. 3(b)), SR1 (Fig. 3(c)), and BR1 (Fig. 3(d)).

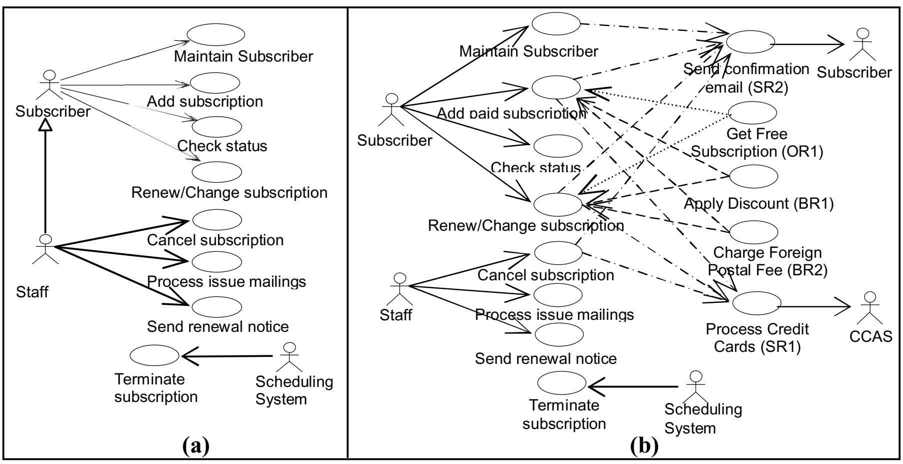



**Fig. 2.** The use case diagrams of SAS. The use case diagram in (a) represents the core functionalities of SAS. The use case diagram in (b) shows the cross-cutting relationships among base use cases and aspectual use cases.

### 3.5 Identify and Model Non-functional Aspects

The only NFR identified from the problem specification is Response time. It is easy to see that this NFR affects almost all the base use cases shown in Fig. 2(b). So it is modeled with the Nonfunctional Aspect Template (as shown in Fig. 3(e)).

### 3.6 The Composition of Aspects to the Base Use Case

Fig. 3 illustrates the composition of the aspectual use cases to the base use case *Add paid subscription*. The composition is based on the definition of *join points* in the

| USE CASE Name | Get Free Subscription **(b)** | |
|---|---|---|
| **Actor** | Subscriber | |
| **Goal** | To Obtain free subscription | |
| **Level** | **Aspect** | |
| **Type** | **Extension** | |
| **Condition** | The subscriber is the author of the articles or the member of the editorial board | |
| **Advice** | Actor Action | System Action |
| | | 1. Displays the subscription data to confirm |
| | 2. Accepts and clicks to proceed | 3. Determines the ending date and renew warning date |
| | .. | . |
| **Pointcut** | Label | Corresponding Join Points |
| | Checking Eligibility | Add Paid Subscription.Check Eligibility for Free Subscription |
| | | Renew/Change Subscription. Check Eligibility for Free Subscription |

| USE CASE Name | Process Credit Card **(c)** | |
|---|---|---|
| **Goal** | To process charges and refunds as related to subscriptions | |
| **Level** | **Aspect** | |
| **Type** | **Inclusion** | |
| **Condition** | The subscriber submits credit card information | |
| **Advice** | Actor Action | System Action |
| | | 1. Send the credit information to CCAS |
| | | 2. Return the successful status flag |
| | | 3. Returns to the primary flow of the base use case |
| **Pointcut** | Label | Corresponding Join Points |
| | Submitting Credit Information | Add Paid Subscription. Submit Credit Information |
| | | Renew/ChangeSubscription .submit Credit Information |
| | | Cancel Subscription. Submit Credit Information |

| USE CASE Name | Add Paid Subscription **(a)** | |
|---|---|---|
| **Actor** | Subscriber, Staff | |
| **Goal** | To obtain a paid subscription of the HOST | |
| **Overview** | … | |
| **Level** | **Base** | |
| **Trigger** | The subscriber has chosen "Paid Subscription" option | |
| **Primary Flow** | Actor Action | System Action |
| | … | … |
| | … | 5. Calculates the total subscription fee |
| | … | … |
| | 9. Enters the name on the Credit Card, Credit Card type, Credit Card No., and expiration date. Then, submit the payment data | … |
| | … | 11. Assigns a subscription ID |
| | … | … |
| **Nonfunctional Aspects** | Response time | |
| **Join Points** | Label | Position |
| | JP1. Check eligibility for free subscription | Before Step 5 |
| | JP2. Submit Credit Information | Within Step 9 |
| | JP3. Confirm successful subscription | After Step 11 |
| | JP4. Check subscription period | Within Step 5 |
| | JP5. Check subscriber's address | Before Step 5 |

| USE CASE Name | **Apply Discount (d)** | |
|---|---|---|
| **Actor** | Staff | |
| **Goal** | To apply 10% discount of subscription fee | |
| **Level** | **Aspect** | |
| **Type** | **Business rule** | |
| **Condition** | The number of subscription year is greater than one. | |
| **Advice** | Actor Action | System Action |
| | | 1. Discount the subscription fee by 10% for the first 3 subscription years |
| | | 2. Shows the subscription period and discounted subscription fee to confirm |
| | 3. Accepts and clicks to proceed | 4. Returns to the primary flow of the base use case |
| **Pointcut** | Label | Corresponding Join Points |
| | **Checking Subscription Period** | **Add Paid Subscription.**Check Subscription Period |
| | | Renew/Change Subscription**.** Check Subscription Period |

| Nonfunctional Aspect Name | **Response Time (e)** | |
|---|---|---|
| **Goal** | To Improve the Performance of the system | |
| **Description** | Information should be accessed and displayed within 5 seconds | |
| **Affected Use Cases** | **Maintain Subscribe; Add Paid Subscription; Check Status; Renew/Change Subscription; Cancel Subscription** | |
| **Relations to other Non-functional Aspects** | **Positive** | **Negative** |
| | .. | … |
| **Priority** | Second | |

Link 1 Link 2 Link 3 Link 4

**Fig. 3.** The composition of aspectual use cases and non-functional aspects to the base use case

base use case and the definition of *pointcuts* in the aspectual use cases (*Get Free Subscription, Process Credit Card,* and *Apply Discount*). For instance, in the aspectual use case "*Apply Discount*", a *pointcut* called "*Checking Subscription Period*" is defined. This *pointcut* corresponds to two join points.

The identification of each corresponding *join point* comprises two parts. The first part refers to the base use case to which the aspectual use case is to be composed (e.g., *Add Paid Subscription*, see **Link 1**). The second part refers to the specific *join point* defined within that base use case (e.g., *Check Subscription Period*, see **Link 2**). Then the position of the *join point* specified in the base use case tells at which specific step within the primary flow the *join point* is defined (see **Link 3**). In this instance,the join point "*JP4 Check Subscription Period*" of the base use case "*Add paid subscription*" occurs "*within Step 5*" of the primary flow. It means that the "*Advice*" of the aspectual use case "*Apply Discount*" is composed to the primary flow of the base use case "*Add Paid Subscription*" at a point within Step 5 in condition "The number of subscription year is greater than one". The base use cases to which a non-functional aspect is composed are referred to through the component of "*Affected Use Cases*" defined in the non-functional aspect template (see **Link 4**).

## 4 Conclusions

This paper proposed a comprehensive aspect-oriented use case method to model complex business requirements. We defined aspects as the requirements cross-cutting multiple use cases. The three-fold contributions of our paper are: First, we extended the types of aspects into four types: NFRs, extending requirements, included requirements, and business rules. Existing approaches have not treated business rules as an aspect. In this paper, we view business rules as an important source of aspects when it cross-cuts more than one use case, since a business rule is usually tangled in many major concerns of a system. Furthermore, it is more changeable than the core functionality. If business rules are represented as aspects, the base model of a system would be disentangled from business rules, so that the base model can evolve independently without invasively affecting each other. This would further improve the reusability of both the base model and the representation of business rules. Second, we proposed three different types of templates for documenting aspect-oriented use case models. With these four types of aspects and three templates, we provided a comprehensive aspect-oriented use-case method to model both functional aspects and nonfunctional aspects. Third, we have shown that the proposed model successfully realizes the composition of aspects to the core functionalities at the requirement level.

## References

1. Araújo, J., Moreira, A., Brito, I., Rashid, A.: Aspect-Oriented Requirements with UML. In: Workshop on Aspect-Oriented Modelling with UML, Dresden, Germany, pp. 1–6 (2002)
2. Araújo, J., Coutinho, P.: Identifying aspectual use cases using a viewpoint-oriented requirements method. In: Early Aspects: Aspect Oriented Requirements Engineering and Architecture Design, Boston (2003)

3. Jacobson, I., Ng, P.-W.: Aspect-oriented Software Development with Use Cases. Addison Wesley, Boston (2005)
4. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C., Loingtier, J.M., Irwin, J.: Aspect-oriented programming. In: Aksit, M., Matsuoka, S. (eds.) ECOOP 1997. LNCS, vol. 1241, pp. 220–242. Springer, Heidelberg (1997)
5. Rashid, A., Moreira, A., Araujo, J.: Modularisation and Composition of Aspectual Requirements. In: 2nd International Conference on Aspect-Oriented Software Development (AOSD), pp. 11–20. ACM, New York (2003)
6. Whittle, J., Araujo, J.: Scenario modelling with aspects. IEE Proceedings 151, 157–171 (2004)

# Exploiting the Complementary Relationship between Use Case Models and Activity Diagrams for Developing Quality Requirements Specifications

Narasimha Bolloju and Sherry Xiaoyun Sun

Department of Information Systems
City University of Hong Kong
Hong Kong SAR, China
{narsi.bolloju,sherry.sun}@cityu.edu.hk

**Abstract.** Use case models and activity diagrams play an important role in the early stages of requirements engineering for systems development. While use case descriptions represent requirements through a sequence of step descriptions in main scenario and alternate scenarios, activity diagrams are often used to connect different use cases and to represent flow of activities corresponding to steps in complex use cases. In the latter type of usage, a complex use case description and the corresponding activity diagram represent a same set of requirements using two different types of artifacts. In such situations, it is necessary to minimize inconsistencies across the models represented by these artifacts and to enhance overall quality of the resulting models. This paper reports the findings from an empirical study aimed at understanding quality dependencies between use case models and activity diagrams, and offers recommendations for developing these artifacts.

**Keywords:** Use case models, activity diagrams, quality requirements specifications.

## 1 Background

Among various types of UML artifacts, use case models and activity diagrams have been found most useful for verifying and validating requirements with client representatives [6]. Dobing and Parsons [ 6] report that use case diagrams and use case descriptions narratives, collectively called as use case models, are among the top four widely used UML components. Use case models drive the development of other important UML artifacts, such as class diagrams and sequence diagrams. While use case descriptions capture detailed steps in main scenario and alternate scenarios, activity diagrams are often used to connect or weave through individual use cases. Although activity diagrams play an important role in workflow modeling [7], this paper focuses on their role in diagrammatic representation of complex use cases [8]. Since the requirements represented by use case models and activity diagrams in such situations are about the same system functionality but from a different perspective, it is

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 144–153, 2008.

possible that inconsistencies in such specifications may arise. It is important to study the relationship between these closely related artifacts, and find methods to enhance their quality in representing requirements.

This paper discusses the complementary relationship between use case models and activity diagrams, and offers recommendations based on an analysis of empirical data collected from several systems analysis projects completed by student teams of an undergraduate course on systems analysis. The following section briefly reviews a conceptual model quality framework that can be effectively applied for evaluating quality of use case models and activity diagrams. Section 3 describes the complementary relationship between these UML artifacts. Section 4 reports findings from an empirical study aimed at understanding quality dependencies between use case models and activity diagrams. Finally, section 5 concludes with recommendations for novice systems analysts in developing these two types of UML artifacts.

## 2 Quality of Use Case Models and Activity Diagrams

Ensuring the quality of the use case models and activity diagrams is an important task considering their role in identifying and specifying the system requirements. Standish Group reports that 46% of the software development projects are still challenged which means that the projects had cost or time overruns or did not fully meet the users' needs [13]. It has been well known that overall effort required for fixing requirements related errors detected in later phases of systems development can be extremely high. Many researchers observe difficulties and complexities associated with using UML (e.g., [1]). Better quality requirement models such as use case models and activity diagrams created in early phases of systems development can minimize communication problems and misunderstandings of requirements, and thereby reduce the effort required for accommodating functionality changes at later stages of systems development.

Quality of use case models and activity diagrams can be studied using conceptual model quality frameworks. Among the mechanisms addressing conceptual model quality surveyed by Wand and Weber [16], frameworks for conceptual model quality provide a systematic structure for evaluation. Several quality criteria and frameworks have been proposed in the literature based on the desirable properties of data models [3]. The framework proposed by Lindland et al. [10] for evaluating the quality of conceptual models has gained wider acceptance through its application in many research studies. An example application of this framework for investigating specific difficulties frequently encountered by novice analysts in developing UML artifacts is reported in [5].

The framework by Lindland et al. [10] uses three linguistic concepts: viz. syntax, semantics and pragmatics, suitable for analyzing the quality of conceptual models. Syntactic correctness of a model implies that all statements in the model are according to the syntax of the language. Semantic quality addresses validity and completeness goals. The validity goal specifies that all statements in the model are correct and relevant to the problem domain. The completeness goal specifies that the model contains all statements about the problem domain that are correct and relevant. Pragmatic quality addresses the comprehension aspect of the model from the stakeholders' perspective.

Studying the quality of use case models from the three perspectives, Bolloju and Leung [5] have identified several quality problems with use cases models such as ambiguous or incomplete step descriptions, improper relationships between use cases, and excessive use of low level steps in use case descriptions.

The quality of activity diagram has been studied in the areas of business process modeling, and workflow modeling [4, 9, 12]. In business process modeling and workflow modeling, activity diagram and its variants are used as a major language to represent the control and coordination of various steps in a business process [12]. To facilitate quality representation of processes, various formalisms have been proposed for verifying the syntactic correctness of activity diagrams and for eliminating errors such as redundancies and deadlocks [4, 9]. An in-depth analysis of erroneous process models reveals that mismatch of splits, joins, branch, and merge is a major contributor for most syntactical errors in process modeling [11]. However, the challenge remains to determine the semantic correctness of an activity diagram, i.e., the appropriate behavior of the process represented by the activity diagram in all scenarios.

Possible inconsistencies between use case models and activity diagrams are likely to increase as use case models and activity diagrams get closely related. The consistencies could arise due to reasons such as mismatch between actors and swimlane labels, representations of alternate scenarios as conditional or parallel paths in activity diagrams, sequencing of steps in use case descriptions and activity in activity diagrams, etc.

## 3 Complementary Relationship between Use Case Models and Activity Diagrams

Table 1 compares the typical characteristics of use case models with that of activity diagrams in capturing system requirements. Although the use case models are expected to drive the development of other UML artifacts, in practice use case models and activity diagrams can be developed in parallel.

The dependencies between use case modeling and activity diagram can be best illustrated through a meta-model (Figure 1). In use case diagrams, actors can be people, external systems, or organizations that play some role in interacting with the system. In activity diagram, swimlanes are used to group activities performed by the same actor. A top-level activity diagram can be created to describe the flow of activities represented by the use cases captured in a use case diagram. In such situation, the activity in the activity diagram under a given swimlane must match the actor(s) associated with the corresponding use case in the use case diagram.

Dependencies between use case models and activity diagram can also occur when an activity diagram is used to show the details of a use case. In this situation, most steps in a use case description along with the post-conditions can then be modeled as activities in the corresponding activity diagram. The use case trigger, the event thatstarts a use case, corresponds to the start activity indicating the state when a process starts in the activity diagram. Also, swimlanes must match the primary actors and supporting actors, which may include other systems, in the use case description.

Note that alternative paths indicated by splits and merges in the activity diagram can be used to capture either the relationships of type generalize and extend between use cases when creating a top-level activity diagram, or to describe the extensions to the basic scenario of a use case when describing the details of the use case in an activity diagram.

**Table 1.** Typical Characteristics of Use Case Models and Activity Diagrams

| **Use case models** | **Activity diagrams** |
|---|---|
| Use case diagram defines systems boundary with activities supported by the system inside the boundary and actors linked to use cases. | Describes process or workflow consisting of activities supported by the system(s) and performed by humans actors. |
| While a use case diagram shows the system scope, use case descriptions capture detailed descriptions of interaction between actors and the system | Activity diagrams usually provide a top-level view of the business process possibly spanning across several actors and systems (can be either top level or detailed). |
| Focus is on the dialog between the primary actor(user) and the system | Focus is on the process flow without identifying specific interaction between the system and user |
| Each use case description focus is limited to a specific type of activity performed by an actor in a relatively short duration (few seconds to few minutes) | Activity diagram captures several activities spanning over long duration (days or weeks or even months) and the activities are performed by one or more systems and actors |
| Use case descriptions use a small set of primitive constructs (sequence and iteration) with step extensions capturing alternate scenarios (exclusive-OR) | Activity diagrams offer a richer set of constructs (sequential, parallel, iteration, decision points including XOR) |
| A use case description cannot be an alternative to an activity diagram because of multiple actors usually involved in the process represented by the activity diagram | Possible to represent any use case description using an activity diagram; however, only complex use cases may actually required to be represented as activity diagrams |
| Emphasize on main success scenario (i.e., most frequently occurring path; without any conditions -- conditions are attached to extensions). | No apparent emphasis on most likely path (though notational conventions might be helpful in showing frequently followed path(s)) |

When a complete set of use case models and activity diagrams are developed to capture all the system requirements, these dependencies become more significant as the overlap in requirements represented by these artifacts increases. If the inconsistencies

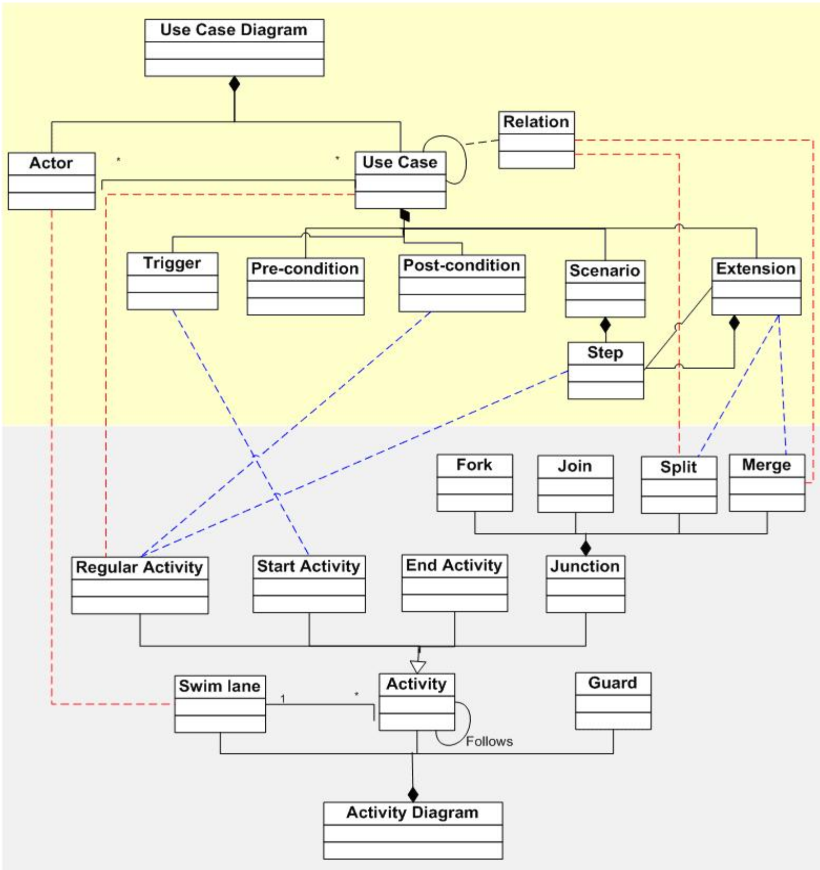



**Fig. 1.** Dependencies between Use Case Models and Activity Diagrams

in requirements captured using such closely related artifacts are not eliminated, the resulting systems may not reflect the users' needs accurately.

## 4 Method

In order to investigate possible inconsistencies among use case models and activity diagrams, this study makes use of a conceptual model quality framework to identify quality problems in closely related use case descriptions and activity diagrams.

For identifying quality problems and relationships among those quality problems, we have used 26 semester-long project work, belonging to six different types of business application areas, submitted by teams of undergraduate students attending a systems analysis course offered at the City University of Hong Kong. Each of these projects has four parts: a use case diagram, a set of use case descriptions, a set of activity diagrams, a class diagram and a set of sequence diagrams corresponding to the use case descriptions. We have excluded class diagrams and sequence diagrams from this study. The students had several opportunities during the semester to improve each part of their project work as they progressed to the next part. Due to this process of teams of students working on semester-long projects with opportunities to revise, we expect that the resulting artifacts are comparable to the work produced by a typical novice analyst.

Coding scheme for assessing the quality of use case models based on Lindland et al's framework [10] has been adopted from a related study [5]. We developed a similar coding scheme for assessing the quality of activity diagrams based on practitioner guidelines available for activity modeling (e.g., [2]) on our experience in teaching activity modeling. Appendix A presents part of the coding scheme used in this study.

Using this coding scheme, we have independently reviewed each project report and identified quality problems in the use case models and activity diagrams. For each type of quality problem identified we have counted the number of occurrences in a given artifact. These numbers varied between 1 and 3 (from our initial observations we found that most artifacts contained one or two errors or a given type). We have also assigned a degree of overlap, an indicator of how closely the pair of artifacts are related, between use case descriptions and activity diagrams in each project report has been estimated on a scale of 1-5 (1 indicates minimum overlap and 5 indicating maximum overlap).

After reviewing all the projects independently, we have randomly selected 3 project reports to verify the reliability of our coding process. In this review, we have noted that 91.5% of instances when one of us has identified a given type of quality problem, the other also had identified the same problem. However, there were differences in the number of occurrences assigned. A thorough analysis of those instances revealed that the larger of the two occurrences was considered more accurate due to the fact that one of us had failed to identify other occurrence(s). IT was also noticed that one was consistently more critical in assessing use case models and the other in activity diagrams. Based on this analysis, it was decided to select the larger number, though it may indicate that the error numbers are inflated, as the more accurate number for each quality problem.

For the subsequent analysis reported in this paper, only the use case descriptions which have a corresponding activity diagram with a minimum overlap score of 3 (moderate overlap) have been selected.

## 5 Findings

These findings are based on quality problems identified in 24 project reports with 55 pairs of overlapping use case descriptions and activity diagrams. Each project report also contained one use case diagram. In this section, we first describe the major groups of quality problems identified (in descending order of their severity based on totals of occurrences identified) and later offer recommendations based on our analysis.

Quality problems found in use case models:

a) Step descriptions: Descriptions of steps, as anticipated, exhibited problems such as using wrong actors; steps with no action indicated or conditions attached; ambiguous due to the absence of required details; and the illogical sequencing of steps.

b) Trigger and pre- & post-conditions: Several specifications of triggers, pre-conditions, and post-conditions in use case descriptions have been found invalid. Some triggers have been found not related to the use case, and some pre-conditions and post-conditions were invalid due to either presence of steps repeating the evaluation of pre-condition or absence of steps ensuring that post-conditions are satisfied.

c) Step extensions: While some of the problems observed were related to inconsistency between a specific step in the main scenario and its extension (e.g., extension is not an alternative to the step in the main scenario), many others of this group were related to missing or incomplete extensions.

d) Types of steps: Problems with granularity of steps (e.g., a sequence of small related steps that can be replaced with one step, a complex step that actually needs to be split into several to provide details), and steps indicating implementation details (e.g., user interface details such as clicking buttons) were found in this group.

e) Notational: Failure to adhere to the syntactic conventions such as proper naming of actors and use cases, symbols used in the diagram, laying out use cases in the diagram based on natural sequence of operations in the real world have been noticed in over half of the project reports.

Quality problems found in activity diagrams:

a) Notational: About three quarter of diagrams had some problem related to usage of indicated notation especially at the merge points. Despite using Microsoft Visio for creating activity diagrams, these diagrams contained invalid or ad hoc notations (e.g., symbols used for flowcharting).

b) Completeness: Many activity diagrams lacked necessary paths or flows (corresponding to the extensions in the matching use case description) required for completeness.

c) Type of activities: Similar to the steps in use case descriptions, problems with granularity of activities and human activities were observed in many diagrams.

d) Layout: Although most use cases studied required only two or three swimlanes, several diagrams were not laid out using swimlanes (instead they resembled flowcharts). In addition, most diagrams did not use explicit merges. Instead, branches originating from splits have been merged implicitly at some activity in most diagrams.

Recommendations for better quality use case models and activity diagrams:

a) Ensure labeling consistency across artifacts: Since the use case models and activity diagrams represent the system requirements from different perspectives, several elements from use case models such as use case names, actor names, and action part of step descriptions are also used in activity diagrams as name of the diagram, swimlane label, and activity names.

b) Enhance structural alignment: In addition to ensuring consistency in naming across various artifacts, structural alignment between use case descriptions and corresponding activity diagrams must be verified. In this regard, relationships among use cases, sequence of steps, step extensions and iterations must be matched against decision points, alternate paths, activity sequence and loops in activity diagrams.

c) Focus on specification quality of step descriptions and activities: Focus on removing ambiguity from step descriptions by enhancing the details presented in step descriptions, removing implementation details (especially when modeling logical requirements), and achieving uniform granularity in steps or activities.

d) Adhere to notational conventions: Adhere to a given set of notational conventions, as true in many types of specifications, plays an important part in enhancing the communication. The tools used for creating these artifacts assist only to a certain extent, and additional guidelines based on our findings can be defined and followed.

## 6 Conclusions

This paper suggests exploiting the complementary relationship between use case descriptions and corresponding activity diagrams for enhancing the quality of those UML artifacts in representing information systems requirements. These suggestions were evolved from an empirical study that is based on a widely used conceptual model quality framework.

The findings from this study offer several implications for practice and research. By understanding various quality problems commonly found in use case models and activity diagrams, methods for training novice analysts can be improved to make them avoid making similar mistakes in representing systems requirements. Artifacts that are free from such common quality problems provide a good foundation for the development of remaining artifacts such as sequence diagrams in structuring the messages, in assigning responsibilities to different objects and in identifying message signatures. Regarding implications for research, this study is the first attempt to address quality of use case models and corresponding activity diagrams, and several extensions are possible. In this study the equivalence of use case steps with activities is established through subjective assessment. Some possible research directions include development of algorithmic support for analyzing inconsistencies in the steps in use cases and activity diagrams; studying differences between overlapping and non-overlapping artifacts; and replicating this study where activity diagrams correspond to a set of use cases linked via include, extend and generalize relationships.

The approach presented in this paper points to an interesting possibility of establishing validation mechanisms where a collection of closely related artifacts are used to represent system requirements from different perspectives. Such mechanisms can be helpful in enhancing the overall quality of requirement specifications and contribute to reducing the effort and resources required for incorporating changes in later phases of systems development due to missed requirements.

**Acknowledgement.** The work described in this paper was substantially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project # 9041217).

## References

1. Agarwal, R., Sinha, A.: Object-oriented modeling with UML: a study of developers' perceptions. Communications of the ACM 46(9), 248–256 (2003)
2. Ambler, S.W.: UML 2 Activity Diagramming Guidelines, vol. 2008 (2007)
3. Batini, C., Ceri, S., Navathe, S.B.: Conceptual Database Design: An Entity-Relationship Approach. Benjamin Cummings Publishing Company, Redwood City (1991)
4. Bi, H., Zhao, J.: Applying Propositional Logic to Workflow Verification. Information Technology and Management 5(3), 293–318 (2004)
5. Bolloju, N., Leung, F.: Assisting Novice Analysts in Developing Quality Conceptual Models with UML. Communications for the ACM 49(7) (2006)
6. Dobing, B., Parsons, J.: How UML is used. Communications of the ACM 49(5), 109–113 (2006)
7. Dumas, M., ter Hofstede, A.: UML activity diagrams as a workflow specification language. In: Gogolla, M., Kobryn, C. (eds.) UML 2001. LNCS, vol. 2185, pp. 76–90. Springer, Heidelberg (2001)
8. Ericsson, M.: Activity Diagrams: What They Are and How to Use Them, vol. 2008 (2004)
9. Eshuis, R.: Symbolic model checking of UML activity diagrams. ACM Transactions on Software Engineering and Methodology (TOSEM) 15(1), 1–38 (2006)
10. Lindland, O.I., Sindre, G., Solvberg, A.: Understanding Quality in Conceptual Modeling. IEEE Software 11(2), 42–49 (1994)
11. Mendling, J., Nüttgens, M.: EPC markup language (EPML): an XML-based interchange format for event-driven process chains (EPC). Information Systems and E-Business Management 4(3), 245–263 (2006)
12. Nysetvold, A., Krogstie, J.: Assessing business processing modeling languages using a generic quality framework. In: Proceedings of the Tenth International Workshop on Exploring Modeling Methods in Systems Analysis and Design (EMMSAD 2005) (2005)
13. Rubinstein, D.: Standish Group Report: There's Less Development Chaos Today. Software Development Times:20070301-20070301 (2007)
14. Siau, K., Cao, Q.: Unified Modeling Language (UML) - A Complexity Analysis. Journal of Database Management 12(1), 26–34 (2001)
15. Siau, K., Erickson, J., Lee, L.: Theoretical vs. Practical Complexity: The Case of UML. Journal of Database Management 16(3), 40–57 (2005)
16. Wand, Y., Weber, R.: Information Systems and Conceptual Modeling: A Research Agenda. Information Systems Research 13(4), 363–376 (2002)

## Appendix A. Excerpts from the Coding Scheme Used for Quality Assessment

| | Use case diagrams | Use case descriptions | Activity diagrams |
|---|---|---|---|
| Syntactic | - Improper labels for actors and use cases<br>- Improper notation used for actors, generalization direction, etc. | - Improper labels<br>- Inconsistency between use case description and use case diagram | - Invalid notation<br>- Deadlock<br>- Multiple starting/ending states |
| Semantic | - Invalid relationship between use cases<br>- Invalid actor in use case diagram<br>- Invalid use case (does not belong to the domain) | - Invalid step in a use case<br>- Invalid extension(s) to a step<br>- Invalid trigger, pre-condition, or post-condition<br>- Missing major extension(s)<br>- Missing an important use case step<br>- Ambiguous use case step description | - Missing guards<br>- Missing decision points<br>- Missing swimlanes<br>- Incomplete diagram (major paths or flows are missing)<br>- Activity assigned to a wrong actor/swimlane<br>- Invalid swimlane label |
| Pragmatic | - Does not use domain specific terminology<br>- Manual operations are listed as use cases<br>- Excessive use of extend<br>- Excessive use of "uses" or "includes" | - Does not use domain specific terminology<br>- Inappropriate step granularity (excessive splitting of steps and combining many steps into one step)<br>- Insufficient detail in use case description<br>- Excessive implementation details | - Diagram layout is poor<br>- Excessive splitting of activities (low level steps)<br>- Does not use domain specific terminology<br>- Diagram is not standardized (e.g., non-standard merge points) |

# Preface to M2AS 2008

Fernando Ferri, Patrizia Grifoni, and Maria Chiara Caschera

IRPPS-CNR, Rome, Italy

This first edition of the *First International Workshop on Modeling Mobile Applications and Services* (M2AS'08) held in conjunction with the *27th International Conference on Conceptual Modeling* (ER 2008) was organized to stimulate the discussion about emerging technological and scientific topics connected to the wide diffusion and the growing demand for advanced mobile devices, their services and their evolution. Mobile devices are becoming central tools for service delivery and future mobile services will very likely come.

That is, models, methods and technologies for Mobile Communication, Mobile Web and services design are relevant discussion issues for providing the state of the scientific and technological research, stimulating new ideas and solutions in the research area. Therefore, the first M2AS'08 Workshop had a strong set of paper presentations. This workshop is marked by a notable scientific maturity and high quality of submitted papers, and the reviewing process was extremely challenging. This year we have had 23 quality submissions from all over the globe. These papers covered a wide spectrum of theoretic and technological issues on emerging areas such as: adaptive services and interaction for mobile devices; mobile systems and architectures; usability methods and models for mobile applications; users' study and application on mobile devices.

After an extensive review process by an international Program Committee of the submitted papers we accepted 9 papers to be presented at the M2AS'08 workshop, and published in the ER 2008 workshops proceedings.

Finally, we offer our thanks to the ER 2008 organization, to all the Program Committee members and to all the authors of the submitted papers for their support and the quality of their work.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, p. 154, 2008.

# A Dynamically Extensible, Service-Based Infrastructure for Mobile Applications

Stefan Kurz, Marius Podwyszynski, and Andreas Schwab

University of Passau
Germany
stefan.kurz@uni-passau.de, podwyszy@fim.uni-passau.de,
schwab@fim.uni-passau.de

**Abstract.** Mobile computing concerns the design and development of applications in highly dynamic and heterogeneous environments to supply the users with all the required services and information. In this paper, we present a dynamically extensible, service-based infrastructure for mobile applications which allow the users to access existing services via mobile devices. As the number of available services is ever growing, we focus on the task of dynamically extending our architecture with new services. Easily adding functionality in a Service-oriented Architecture (SOA) is a matter of clean design decisions. However, if these services are to be accessed by already deployed client applications, there is a need for doing so without high recoding and redistribution costs. While discovering newly available services is a solved problem, dynamically integrating the new functionality (adding GUI, local data bindings and controls) is a non-trivial task. We propose a generic mobile client application automatically extending its functionality when new services become available.

## 1 Introduction

Current information and communication technology provides multiple ways to access information systems and to use services made available via the Internet. Traditional personal computers are only one of the many ways to do so. As traditional computing technology is becoming more and more mobile, also different mobile devices like PDAs and cell phones gain the ability to access information resources and services. It seems to be a logical consequence that mobile infrastructures may fundamentally change the way people use information systems: access to systems is possible from any location ("anywhere"), at a moment's notice ("anytime") and in a multiple modality ("anyhow").

In this paper, we assume a highly dynamic and heterogeneous IT system environment, e.g. of a university. Various systems are available and provide users with a huge variety of information resources and services: e.g. a Learning Management System (LMS) that supplies students with current course information and which offers an e-portfolio that allows them to document their e-learning activities. The e-portfolio in turn may comprise the students' transcripts of records whose management is in the focus of a Campus Management System (CMS). As

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 155–164, 2008.

these systems (e.g. LMS and CMS) offer services (e.g. e-portfolio) that would surely increase in acceptance and usage if accessible from anywhere at anytime in different ways, it seems to be a good idea to extend the traditional access via personal computers by mobile access via mobile devices. In this paper, we present a dynamically extensible, service-based infrastructure which allows users to access the existing services of an IT system environment (like the one sketched above) via a mobile application. This application can be seen as a mobile information system which according to [5] can be defined as an information system that provides access to information resources and services through end-user terminals that are easily moveable, operable at any location, and typically provided with wireless connection. The main goal of the application is to support the user's mobility. The degree of the mobility coincides with the application's independency of the server architecture. The more autonomous the client application is, the less is the user impaired by missing availability of the server architecture (e.g. if no connection can be established).

There are different possibilities to realize the mobile application. The seemingly easiest way to do so is a web application that can be accessed by the Web browser of the mobile device. However, we propose a stand-alone client application which can be deployed to the mobile device. This approach offers the major advantage that the client application can be used even if no connection to the server architecture is established; locally held data is permanently and quickly accessible by the user. Unfortunately, on the other hand, a deployed stand-alone client is difficult to extend when new services become available (i.e. GUI building, data binding etc. has to be done). To overcome this drawback, we focus on the task of dynamically extending our infrastructure with new services and of dynamically integrating this new functionality within our client application.

The remainder of the paper is organized as follows: in section 2 two use cases are used to derive goals and requirements of our approach. The developed architecture (server and client side) is outlined in section 3. Section 4 focuses on the dynamic extensibility of the infrastructure (again server and client side are considered). The resulting workflow of deploying a new service within our architecture is sketched in section 5. To indicate the practical applicability, section 6 very briefly describes an example we used to evaluate our approach. The paper ends with an overview of related work in section 7 and a conclusion in section 8.

## 2 Goals and Requirements

In the following, we sketch two use cases that should illustrate a scenario in which our approach could be set up. As an example, the used scenario is situated in an university environment.

*Use Case 1: Retrieving current information.* Student $S$ is sitting in the cafeteria and not sure about the facts when and where the next course is taking place. $S$ starts our mobile application at his or her cell phone and invokes the timetable service to find out about the individual course information which is updated frequently.

*Use Case 2: Integrating new services.* The course has ended, and while $S$ is going back to the cafeteria, the mobile application discovers the newly available library service that informs users about the status of book orderings. $S$ is asked if this new service should be integrated. $S$ accepts, and after some seconds the mobile application is able to communicate with the library service.

From the use cases above, the following basic goals and requirements our approach should satisfy can be derived:

1. **Integrated Systems and Data.** As suggested in Use Case 1, available systems have to be integrated to act as data providers within the mobile infrastructure.
2. **Standardization and Interoperability.** The integration of the underlying systems should be done in a standardized way: a deployment as web services offers standardized communication and allows setting up an infrastructure based on the principles of a SOA. Moreover, the interoperability between services and different mobile operating systems is maintained.
3. **Extensibility and Generic Communication.** As already mentioned (and illustrated in Use Case 2), extending the infrastructure with new services and adapting the mobile application accordingly is a major goal of our approach. The mobile application must be able to discover and dynamically integrate newly available services, thereby automatically extending its functionality.
4. **Scalability and Performance.** As data transmissions via a wireless connection might still be expensive in time and costs, the number of requests as well as the amount of transmitted data should be minimized.

## 3 Architecture for Extensible Service-Based Mobile Applications

The dynamic nature of a SOA has motivated our approach of an *Architecture for Extensible Service-based Mobile Applications*. The main components of our architecture are deployed as web services and a major part of the communication flow is taken by invoking customized web services deployed within the SOA. Moreover, the lose coupling, interoperability, and reusability of web services renders our architecture highly dynamic and extensible.

The Architecture for Extensible Service-based Mobile Applications is illustrated in fig. 1 for a SOA of the university scenario.[1] Platform independent mobile clients (which are depicted in the top left corner of fig. 1) communicate with services of the SOA over an encrypted Internet connection via an intermediate service, the so-called *Communication Service*. The proposed SOA is separated into the two logical layers Base and Super. *Base Services* are elementary services which perform basic activities at the underlying legacy systems in terms of CRUD[2] access to a database or the spooling of print jobs. Furthermore, Base Services achieve a high level of reusability for other clients by providing a low-level

[1] For more details on the setting of our architecture, see also [6].

[2] The acronym CRUD stands for ***C**reate, **R**ead, **U**pdate, **D**elete* - the four basic functions of persistent storage.

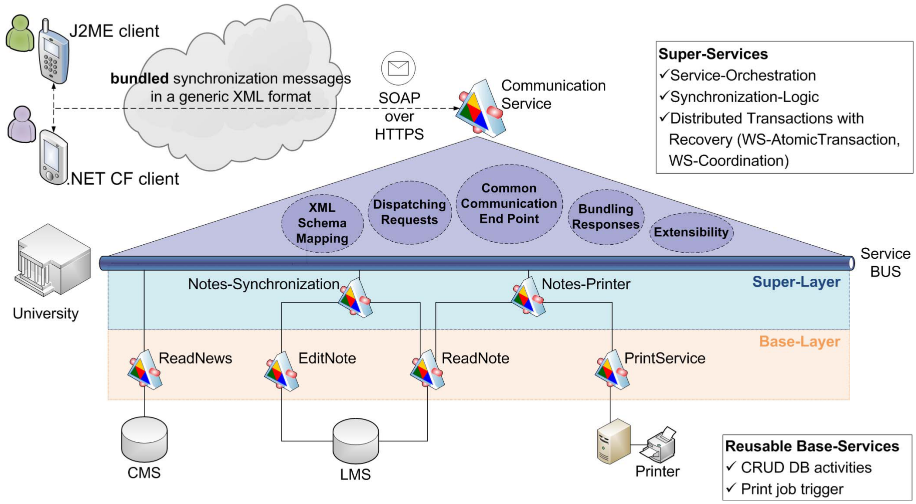



**Fig. 1.** Architecture for Extensible Service-based Mobile Applications

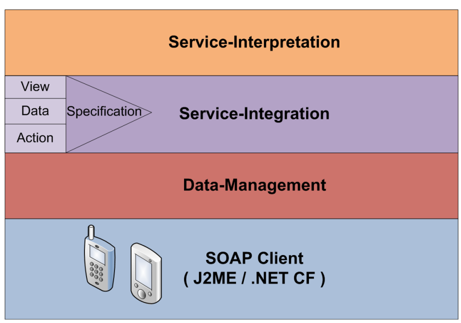



**Fig. 2.** Dynamic Mobile Client Framework

business logic. *Super Services* are identified by sophisticated service orchestrations. As compared to Base Services, Super Services handle more complex tasks in a use case-dependent manner (e.g. data synchronization or distributed atomic transactions; for the former, see below). The *Service BUS* connects all deployed services of any logical layer to the Communication Service and therefore enables the mobile clients to access them as well.

The components of the proposed *Mobile Client Framework* are presented in fig. 2. Starting at the bottom, the framework is written either in Java or C# and provides a *SOAP*[3] *Client* to handle the web service invocations of the Communication Service. The *Data Management* layer provides the ability to cache and store data from a service persistently on the mobile device. The layers *Service*

[3] Simple Object Access Protocol (SOAP), http://www.w3.org/TR/soap/

*Integration* and *Service Interpretation* are described in more detail in section 4 as they are the core components in achieving client-side dynamic extensibility.

As an application example a data synchronization capability has been implemented. The synchronization logic is located at the Super layer. For each synchronizing Super Service we set up services for reading and updating data at the Base layer which are invoked in a predefined sequence. The synchronization logic may be described by the following four steps:

1. obtain new data by invoking the reading service,
2. check for data conflicts in the returned data,
3. invoke the updating service if there are no conflicts in this data,
4. combine the results from the steps 1. - 3. into a final response.

Following a timestamp logic, the data transmission to the mobile client can be limited to the delta data only (cf. requirement 4 in section 2).

# 4 Dynamic Extensibility

To achieve the dynamic extensibility of our architecture, both the server side and the client side have been designed in a way that enables and supports adding new functionality as described in the following. In the next section, we then sketch the steps to be done deploying new services within our architecture.

## 4.1 Server-Side Design Decisions

The server side uses the concepts of SOA and provides a generic way for mobile clients to access the deployed functionality.

**SOA.** Using a SOA renders the functionality of the server-side information system highly extensible. This is due to the main features of a Service-oriented Architecture as are contract-based communication, lose coupling, the reusability of services and their orchestration [3]. Implementing elementary Base Services elevates the chance to reuse their functionality, e.g. by orchestrating several Base Services in different workflows or application contexts.

**Generic Communication Endpoint.** However, if we have a SOA deployed with its growing number of contract-based public interfaces, we need a simple way to access all these services by mobile clients. To keep the client-side framework independent and avoid recoding costs, we propose the adoption of a single communication endpoint to connect the generic framework functionality of the client with the specific services within the SOA.

Assume a request which is initiated from a mobile client as illustrated in the top left corner of fig. 1. The single communication endpoint for the client is represented by the Communication Service which is deployed as a Web Service and provides a *generic communication interface* allowing the mobile client to communicate with each server-side service in a common modality. Thus, the client can send a generic synchronization request to the SOA containing a list of generic service requests. At the Communication Service, each listed service request is dispatched to an appropriate *communication adapter* which maps the request and response parameter types of a service's WSDL document to the generic XML-Schema of the communication interface. This mapping enables a communication adapter to transform a generic synchronization request into a service-specific request, to invoke the appropriate service with the transformed request as parameter, and finally to transform a service-specific response into a generic synchronization response. While each communication adapter runs in its own thread to enhance concurrency among service invocations, there is only one instance of the Communication Service which waits for all communication adapters to return and to bundle their service responses into a single generic synchronization response message that is then returned to the mobile client.

**Service Specifications.** Enabling the mobile client to subscribe to new services and to integrate new service functionality is done by providing a common service description. This service description contains information about the GUI, the structure of cached data and the mapping of specific server-side functionality to generic client-side functionality. The service's WSDL document provides information about the objects that are exchanged by the service. This information can be reused to create partial service specifications for the mobile application. Each service has to provide specifications for data, view and action. A *data specification* declares the structure of cached data. How this cached data is displayed at the client, i.e. the appearance order and listing type to display several items (e.g. tree, list, or tab view), is defined by a *view specification*. An *action specification* describes the action that may be performed at the service and is mapped to a CRUD action at the mobile client, which is executed on the cached data.

### 4.2 Client-Side Design Decisions

The dynamic extensibility of the mobile client application is illustrated by the concepts *Service Integration* and *Service Interpretation* which have been implemented as layers of the mobile client framework (cf. fig. 2).

**Service Integration.** When a new service becomes available and the user decides to subscribe to it, the service has to be integrated at the mobile client before it can be used. The view, data, and action specifications for the new service are obtained and stored locally at the mobile client. Finally, the cache tables for the mobile data management are created by interpreting the data specification. After the mobile service integration has finished successfully, the new service is ready to use at the mobile device.

**Service Interpretation.** When the user decides to view data received from a service, the service interpretation handles the graphical and functional representation of this data. The view specification is interpreted to create empty view templates which are subsequently filled with the locally available service data on-the-fly. Multiple data from a service is appended either as a tree node, a list element or in another tab, depending on the defined specification. Such default views may be extended to a composed one, when data from multiple services is bundled into one view. Moreover, a graphically coherent view with a 1:n-data-relationship can be generated. As an example, fig. 4 shows GUIs that have been generated at the mobile client on-the-fly from data of several course services according to their specifications. The arrows between the screenshots indicate the user's selections in a view. Moreover, the action specification is parsed to structure the mobile application's dialog flow. This fact enables the application to decide according to the specification which view is displayed next after a local action has been performed. Furthermore, the mobile client's data operations are extended by the mapped CRUD functionality of a service that is provided by the action specification of the service.

## 5 Workflow for Dynamic Extensions

Having set up our extensible architecture, a service provider has to perform the steps depicted in fig. 3 to publish a new service.

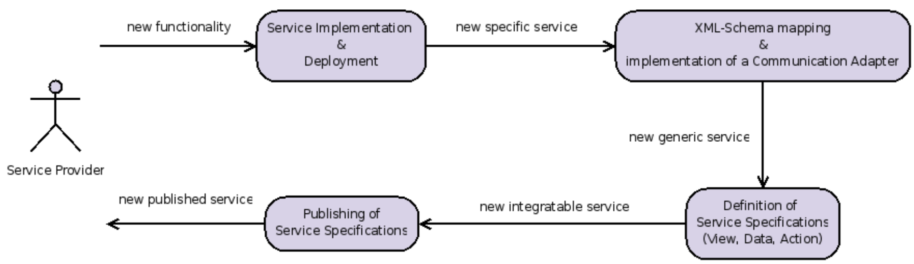



**Fig. 3.** Workflow for Dynamic Extensions performed by a Service Provider

First of all, the desired functionality has to be deployed as a new Web Service within the SOA (if possible and required, the new functionality can be built upon already existing services).

The service's WSDL document builds the basis to next implement the communication adapters, which map the service's specific objects to common objects that can be sent via the generic communication service (XML-Schema mapping).

Finally, the service specification (consisting of view, data and action specifications) has to be defined and published within the architecture at a dedicated integration service which manages the specifications of all available services and which can be queried by the mobile client application for discovering newly available services.

## 6 Our Approach in Practice

In our prototype implementation, the Course Management System *Stud.IP* [9] is used to administrate and publish the university calendar. The published course

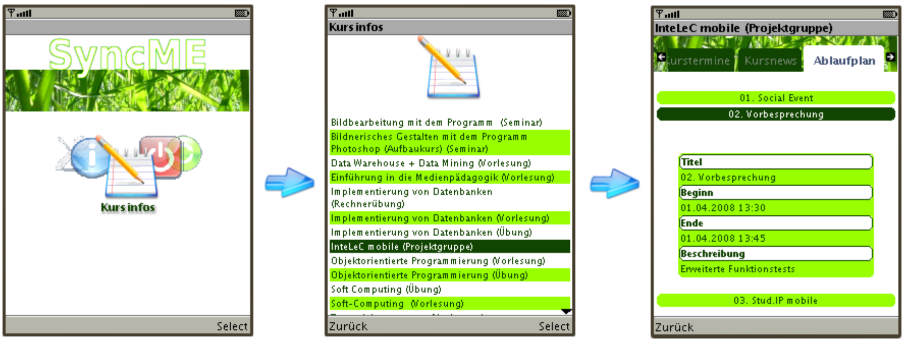



**Fig. 4.** Customized mobile application with generated GUIs for the main menu, a list of courses and detailed course information

information contains but is not limited to detailed course descriptions, schedules and appointments, files, and news. We ported these functionalities to mobile applications by setting up the proposed infrastructure in the university IT system environment. Communication adapters for the services course details, course dates, course news, and course files have been created along with the customized service specifications. The Communication Service has been deployed as a Java web service in the Sun Java System Application Server accessible from any mobile client at the campus via the university's WLAN. Currently, we are setting up a testing of the mobile application by a greater number of users.

## 7 Related Work

The development of mobile frameworks has been considered in several other projects. Concerning the synchronization capabilities of our architecture the best similarities can be recognized in the SodaSync [2] and the SmartDOTS [4] frameworks. However, *SodaSync* forces possible data providers to run in a SDO-supported environment handling all business data as Service Data Objects (SDO). The *SmartDOTS* framework is a context-aware synchronization and replication framework for mobile devices. It supports the modeling of business data which sets context dependent parameters like target device, network selection or task. Synchronization conflict resolution is currently work in progress at the SmartDOTS project. In contrary, our infrastructure benefits from the platform independency of data providers which are deployed as interoperable web services. Moreover, we implemented a synchronization conflict detection logic at the server side and the possibility to resolve the conflicts at the mobile client. Mobile data persistence is available in SodaSync, SmartDOTS and our mobile client framework. However, we are not limited to synchronized data access but also handle complex data interpretations.

Regarding the data transformation and presentation at the mobile client, the *Mobile Delivery Server (MDS)* [1] uses similar techniques to ours. The MDS acts as a middleware server between a LMS and mobile devices. In this approach, content from the LMS is transformed into mobile content respecting device limitations; so large files like videos are not directly sent to the mobile client. At the mobile device a content player is used to display the retrieved content. The MDS lacks of the possibility to extend the mobile clients dynamically after distribution since new LMS content requires the compilation and redistribution of a new content player for the mobile devices. In contrast, our mobile application is able to generate a data representation on-the-fly and to integrate new services dynamically.

The *M-Services Framework* [7] proposes a Web service-based architecture for providing access to services from mobile devices. It defines service managers which act as a mediator layer between the service providers and mobile devices. This concept is similar to the Communication Service in our infrastructure. However, we do not support service discovery via UDDI like the M-Services Framework. Furthermore, we provide communication adapters and service specifications which cover more detailed information than is contained in the service's WSDL documents. In the M-Services Framework, new services may be integrated at the mobile device in a generic way by parsing the provided service's WSDL document. This fact impedes that the framework generates advanced views for interrelated data like in our university scenario. In addition, the M-Services Framework provides no mobile data synchronization, management, or storage capabilities which is essential for mobile offline data access which was an early requirement of our scenario.

## 8 Summary, Conclusion and Future Work

With the results of our work it is feasible to dynamically integrate systems of an IT infrastructure into a mobile environment: newly available services offered by these systems can be deployed as Web services to our Service-oriented Architecture. Our mobile client application discovers these services and automatically extends its functionality by importing the service specifications: data bindings and controls are added and the appropriate GUI is built on-the-fly by interpreting the view specification. The communication between client application and service architecture is done in a generic way so that each service can be integrated and handled uniformly. We evaluated our results within the IT infrastructure of our university: several services have been deployed and made accessible for our client application, which is available for J2ME and Windows Mobile platforms.

However, there are still several open issues: currently, we are working on further improving the specification and deployment process of services. On the one hand, service specifications can already be interpreted on-the-fly to build a GUI. On the other hand, this GUI building process does not consider the requirements and constraints of the device (e.g. display-width constraints). We propose a MDA-based approach which allows the specification of views, data and actions with the means of UML models. The modeling starts from an platform independent model which is transformed according to specifics of the target platform

(J2ME or Windows Mobile) as well as to specifics of the target device (requirements and constraints). The resulting model represents a device-specific service specification which then can be sent to the mobile client [8]. Furthermore, we are planning to enhance local persistent storage by using a mobile DBMS.

## References


1. Averberg, D., Boppert, J., Holzweibig, K.: Mobile Delivery Server (MDS) - A Solution for Resolving Problems and Limitations in Mobile E-Learning Scenarios. In: International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL 2006). IEEE DL Publication, Los Alamitos (2006)
2. Castro, P., Giraud, F., Konuru, R., Purakayastha, A., Yeh, D.: A Programming Framework for Mobilizing Enterprise Applications. In: Proceedings of the Sixth IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 2004), pp. 196–205 (2004)
3. Erl, T.: Service-Oriented Architecture. Concepts, Technology, and Design. Prentice Hall International, Englewood Cliffs (2005)
4. Kurschl, W., Mitsch, S., Prokop, R.: SmartDOTS - A Framework for Efficient Data Synchronization on Mobile Devices. In: Proceedings of the Third International Conference on Information Technology: New Generations (ITNG 2006), pp. 300–305 (2006)
5. Pernici, B.: Mobile Information Systems: Infrastructure and Design for Adaptivity and Flexibility. Springer, New York (2006)
6. Podwyszynski M., Schwab A.: Design and implementation of a mobile J2ME/.NET application and a transactional SOA to synchronize students' e-portfolios. Bachelor Theses, University of Passau (2007)
7. Sanchez-Nielsen, E., Martin-Ruiz, S., Rodriguez-Pedrianes, J.: An Open and Dynamical Service Oriented Architecture for Supporting Mobile Services. In: Proceedings of the 6th International Conference on Web Engineering (ICWE 2006), pp. 121–128 (2006)
8. Schwab, A.: Generierung von Bedienoberflächen für mobile Clientanwendungen in einer service-orientierten Infrastruktur. Diploma Thesis in German, University of Passau (2008)
9. Stud. IP: Project Description and Documentation, http://www.campussource.de/org/software/studip/

# The Situation Lens: Looking into Personal Service Composition

Augusto Celentano, Stefano Faralli, and Fabio Pittarello

Università Ca' Foscari di Venezia, Dipartimento di Informatica
Via Torino 155, 30172 Mestre (VE), Italy
{auce,faralli,pitt}@dsi.unive.it

**Abstract.** In this paper we discuss about composing services from standard suites of applications for personal data management in mobile devices. We propose a data model and an interface model that allows users to navigate among services according to the situation evolution as perceived and interpreted by the users themselves. The interface model acts as a lens exploring the situation, zooming into the details, covering different areas of the personal data, supporting the user in the role of composer of personal services.

## 1 Introduction

Service composition is usually approached as a problem to be solved automatically: environment sensing, service discovery and task planning are the key terms driving the search for efficient ways to compose black-box services, building applications adaptable to the user needs. The Activity Theory concepts [1] play a big role in such a view: the needed services are related to a user state that can vary only within defined boundaries, which do not influence his/her activities (hence his/her goal), but only the way in which the tasks are executed.

This view falls short in many activities related to personal data management, which evolve in space and time as the user proceeds in executing the tasks. The scenario is closer to the one defined by the Situated Cognition approach [2]: actions are driven by the user situation, a complex set of variables related not only to space, time, profile and environment, but also to the user goals, history, emotional status, etc.. The situation evolves as new information is acquired and results of previous tasks collected. In practice, users start by defining a coarse-grain plan that is continuously refined and adapted, evolving according to the many variants in the situation encountered during its execution [3].

In this paper we discuss about composing personal activities supported by information stored in portable devices like UMPC, PDA, smartphone, etc. Such devices offer standard suites of applications for personal data management: agenda, address book, mailing and messaging, maps, etc., often independent or loosely coupled, used with different skills according to the personal style of the user. Such applications compel the user to find own way to compose and integrate the services offered by the device by activating sequences of commands related

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 165–174, 2008.

to different applications without the benefit either of an adequate information flow, or of a smooth transition from one application to the other.

We propose a model for user task organization and an interface model that links the personal services provided by a personal information management (PIM) suite, allowing users to navigate among them according to the situation evolution as perceived and interpreted by the users themselves, hence not driven by predefined adaptation mechanisms. The interface model acts as a *lens* exploring the situation, zooming into the details, covering different areas of the personal data, supporting the user in the role of composer of personal services.

## 2 Related Work

Situated computing is based on the idea that the different parameters that define the user situation may change, requiring a reorganization of the information provided and of the accessible services. The word *situation* is replacing the word *context* in today literature [4], moving from physical, observable features of the user environment to a higher level of context including the user goals, plans, activity and history [5]. Situated computing derives from Situated Cognition [2], a behavior model claiming the dependence of the user actions on the set of environmental, social and psychological variables describing the user situation. It has been studied as opposed to the Activity Theory model [1], based on the definition of action plans to reach anticipated goals. The two action paradigms are not completely disjoint, and relations between them have been studied [3,6].

Situated computing is strongly related to the concept of adaptation of information and interaction, and exploits in mobile computing its full potential, even if its importance is not restricted to a continuously changing context. Context modelling, adaptation and interaction are the three keywords denoting the ability to design systems able to perceive the user status, to provide proper information and services, and to offer a user oriented effective interaction style.

In information processing applications, namely in the area of personal databases, closer to the scope of this paper, an interesting approach to context adaptation is the *Context-ADDICT* project (http://poseidon.elet.polimi.it/ca/), aiming at selecting the part of a database relevant for the context of mobile users, through the dynamic hooking and integration of information sources [7,8,9]. Adaptation is based on an ontological representation of the application domain, and of data source contents, tailored on the user context dimensions. Differently from our approach, focused on the user interface level for selecting different perspectives on data, the Context-ADDICT system is targeted to design time, and its scope does not include the user interface and the interaction functions.

In the area of service computing, adaptation has been investigated primarily to find the best set of services to build an application fulfilling a user need. Architectures and strategies for context-aware service composition have been investigated in [10,11,12]. As noted in Section 1, their primary focus is to automate service discovery and composition, while our proposal puts the user in the center of the service composition choices.

User interaction in mobile systems has been investigated under the perspectives of usability and information presentation, supporting the user to seamlessly and naturally navigate through information and services. Zoomable user interfaces (ZUIs) and gestures improve the usability of small devices applications [13,14]. Zooming technology is becoming popular in new portable systems, like Microsoft WM 6.0-based devices and the Apple iPhone/iPod. SDKs and third part APIs, like the *Piccolo* framework (http://www.cs.umd.edu/hcil/piccolo/), provide standard widgets and gesture recognition modules and, in some cases, allow software developers to easily extend the framework with custom provided components. The applications built with such interfaces, while sharing with our proposal the smooth transition between overview and detail view on user data and services, do not help the user to change the perspective while browsing data. According to our vision, the system should allow the user to select information and services belonging to a category and present, on a synthetic view, the related items belonging to other categories. Each item should be accessible in a detail view or used as the focus of a new perspective in a highly dynamic fashion.

## 3 Situated User Activities

The management of personal activities during day by day professional life is a good example of a mixture of the Activity Theory and the Situation Cognition approaches. In planning such activities a person flows through a set of information that are in part objective (facts, regulations and practices), in part subjective (judgements, feelings, history, experience) and builds an initial version of a plan. The more the activity is clerical, the more the plan will survive to new information. However, decision based activities typical of managers and professionals are less stable, and need a higher level of adaptation to changing situations. In such cases an initial plan is roughly sketched and is adapted and completed as the activity evolves. Information search is an essential part of such activities; information access is not regulated by a series of independent, atomic needs, but by a complex network of interrelations that link the activity execution to the situation changes.

Using popular terms, information is scarcely *queried* but is mostly *browsed* according to the user context, which is often known (or sensed) by the user but can hardly be formalized. Browsing, however, is not completely free, since it is driven by the evolving situation, therefore based on previously acquired information. The result is a controlled navigation among services and information, where the user decides on the fly which are the relevant paths to follow for fulfilling the activity tasks.

This type of user behaviour assigns to the user the role of *service composer*, designing a scenario very different from (almost opposite to) the automatic service discovery and composition of pervasive computing. The system may help the user evidencing the connections between people, locations and services belonging to separate activities. The knowledge of overlapping elements (e.g., a specific location shared by the current activity and by other activities) may become relevant if the situation changes; such elements, if properly evidenced, may act as switching points that can suggest the user to begin another activity.

The active role of the user in service selection, in our vision, is not a negative feature but a necessity, due to the variability of the cases the user needs to face. Three important issues stem. First, our scenario is sampled on a user performing activities related to personal information management and day by day organization of own professional and social life. Both are subject to a reasonable overview planning but must adapt to events and constraints set by other people with whom the user relates. Second, in a PDA-based information management services are general purpose and largely independent, even if collected in suites. This is due to technical limits of program integration in small capacity devices and to the need of supporting a wide spread of use habits. Third, a considerable part of information needs is resolved by searching the Web, and the retrieved information is used in the services useful for performing activity related tasks.

In the personal information management framework the services available to a user are split in two classes: generic and specialized services (i.e., programs). Generic services like agenda, address book and messaging, in principle are not related to the fulfillment of a specific task but are used as commodities, much as generic libraries in a programming environment. Specialized services are in principle related to the fulfillment of a task, such as map-based information discovery. The distinction is fuzzy, and double-faced services are common: for example, mail can be considered a generic services when reading new mail (since the content is unexpected), but is a specialized service targeted to a specific use when replying to a message, or when writing a new message related to a task.

A system supporting personal activities should:

1. Know about user activities and tasks, freely set by the user, possibly according to templates that represent initial, modifiable, activity plans.
2. Track the status of tasks and activities, and track the set of information associated to the tasks.
3. Present to the user a set of information and services related to the user *situation* as described by the activities and tasks status: services strictly functional to a task could be initiated by the system by suggesting an execution order, and by filling variable data with information taken from the situation.
4. Display the relations between the activities and the tasks defined by the user, that might have been defined at different times. Merging data related to different situations, with the explicit synthetic representation of the contact points, is useful to give a higher level of knowledge, mostly effective if the situation changes.

We propose to support this type of operations with a model, called *situation lens*, composed of a schema (a simplified ontology) for describing activities, tasks, services and situations, a mapping between the schema and the personal information management system, and an interface able to explore the information and the services contributing to the user situation, zooming into the details, covering different areas of the personal data, thus supporting the user in the personal composition of services.

## 4 The Situation Model

The model of activities and services on which the situation lens is grounded is close to the *situation metaphor* defined by Hewagamage and Hirakawa [5]. It is built around four classes of objects: activities, tasks, events and services. In the following, details are omitted where not relevant in the scope of this paper.

**Activity.** An activity is a set of tasks, $act = \{t_1, t_2, \ldots\}$, partially ordered. The set may change during the activity execution: new tasks can be added, some tasks can be removed, thus making the activity to evolve from an initial state $act_0$ to a final state $act_f$. The initial state represents the user needs as defined at activity planning, and the set of tasks is both a plan for the activity and its history, when tasks are done. The initial state of an activity can be empty, semantically representing a placeholder for an activity anticipated by the user but not planned at all.

**Task.** A task is a tuple $t = \langle task\_name, description, ES, status \rangle$. *Task_name* and *description* have obvious meanings; $ES$ is a set of pairs $\langle e, s \rangle$, meaning that upon occurrence of event $e$ service $s$ can be executed; *status* is an enumerative value ($to\_do, in\_progess, done$) that can be set by service execution.

**Event.** An event is a tuple $e = \langle event\_name, time, location, contact, message \rangle$. An event can occur at a specified time, during a time interval, when reaching a location, upon a message arrival, or can be associated to a contact, e.g., adding a new contact triggers the event.

**Service.** A service is a tuple $s = \langle service\_name, description, A \rangle$, where $A$ is a set of actions: *execute_service*, *create_event* and *change_status*, with obvious meaning. Services are executed by programs through which the user accesses personal data, creates and sends messages, adds contacts, writes notes (which are the typical applications of a PDA program suite), and accesses external services, e.g., in the shape of an URL fed to a browser.

**Situation.** The data instances describe a state of the user activity. Instances taken at different time stamps describe the evolution of the user activities. A snapshot of the data instances is a *situation*, the description of the information the user has about the current status of his/her activity. Snapshots taken at different times describe the user history, as in the Hewagamage and Hirakawa model [5].

It is worth to note that in the above model events and services are prototypes to be refined on the actual PDA software environment. In the definition of an event, time and location define at the minimum extent the objective user

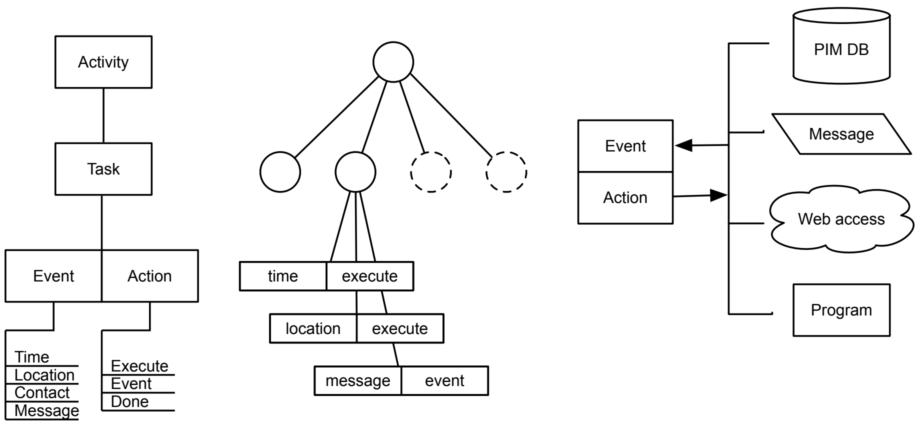



**Fig. 1.** The situation lens model

situation, while contacts and messages are the minimal interface with external information, defining the subjective user situation. Other components may extend the model, but do not add relevant concepts for the model goals. The same holds for service actions, which are described as a core that can be extended at need.

The model defines a set of classes from which data instances can be derived, linked to services and information managed by the user applications. In Figure 1 the activity and service model is on the left. The centre diagram depicts the instances: tasks to be completed are represented with dashed borders; task instances are event-action pairs, where actions can generate new events, which, in turn, drive the user to add new tasks. On the right, the PIM Database collects data about user contacts, agenda, notes, maps, etc.; messages, web access and external programs represent the prototypes of services the user can activate to fulfill his/her activity; events are triggered by information contained in the PIM or by the user intervention, who interprets the messages, the browsed information and the result of the executed programs.

## 5 The Situation Lens

The *situation lens* is a dynamic filter on the data managed by the PIM, and on the available services; it generates a situation snapshot according to three different perspectives:

- a *global perspective*, showing a synthesis of the user activity: a qualitative view of the activity status and of the entities involved is displayed as an index to explore the activity components;
- a *task perspective*: the part of the user situation related to a specific task is given, acting both as a summary of the task and an index to the task details;

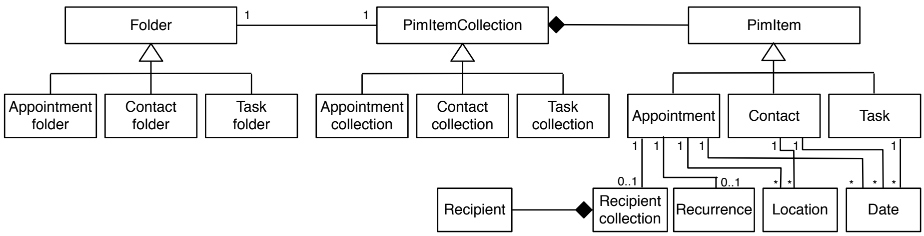



**Fig. 2.** The Pocket Outlook Object Model

– a *detail perspective*, allowing the user to explore detail data extracted from the personal information database and to execute programs and services.

In the following of this paper we assume as the underlying platform the Windows Mobile 5 system. The discussion is however general and largely independent from the platform.

### 5.1 Mapping the Situation to the Services

The PIM in Windows Mobile systems includes several applications that can be viewed as services offered to the user: address book, agenda, messaging, tasks. They are activated as independent programs and refer to information chunks that are independently stored. However, during the execution of an activity, the user switches from a service to another building in his/her mind a map of related data elements. For example, upon receiving a mail message from a colleague, the user might decide to call her instead of replying with a mail message, switching from the mail program to the address book, indexed on the name of the mail sender. In the same way, when reading about next meeting in agenda, the user could decide to consult a map of the meeting location, e.g., to find a fast way to move there, indexed on the address of the colleague who has organized it.

Figure 2 shows a fragment of the *Pocket Outlook Object Model* (POOM), implemented in Windows Mobile systems. The model is strongly based on the concept of *collection* for representing and storing the data instances. As such, while operations to manage items are easy and similar (e.g., adding, removing and sorting items), the relations among data are not apparent from this schema. Indeed, only the right part of the diagram introduces classes and attributes for the data relevant for the user (the *Appointment*, *Contact* and *Task* entities and their sub-entities).

In the situation lens framework, the situation is mapped to the POOM through an extension of the entity *Task*, which in the POOM is simply a description with a status and possibly a deadline. Indeed, the POOM task is similar to an action in the Situation Lens model. In our model tasks are containers of references to other PIM items, collecting the information corresponding to keys to contacts, appointments, actions, messages, and possibly other information like maps. Figure 3 shows an example of the *task* and *detail* perspectives. The screenshot on the

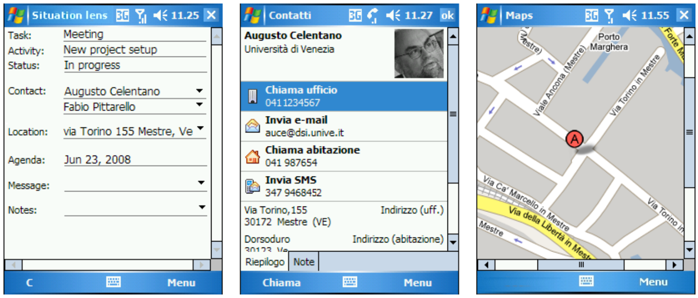



**Fig. 3.** The task and detail perspective interface

left shows the task description, which is an index to detail information. Tapping an item opens a detail perspective view in the PDA service that creates and manages that type of data. The screenshot in the center of Figure 3 shows the information of one of the two contacts of the task, while the right screenshot shows a map centered on the location associated to the task. The PDA services activated show their own interface and behavior, as if opened directly from the *Program* menu of the PDA.

Switching between services preserves the situation since service activation is always done through the task description, which filters the instances of data accessed by specific services. Generic services can be activated from the PDA program menu.

## 5.2 The Situation Lens Interface

Figure 4 illustrates the interface of the situation lens *global perspective*, giving the user an overview of the situation related to an activity. The interface has been designed with two goals in mind: (1) to provide an immediate perception of the task complexity through a summary view of the entities involved, and (2) to allow the user to explore the relations between the entities and the relations with other tasks and activities, masking the details.

The four screenshots show different views on an activity or a task. The first screenshot shows the interface structure. The situation lens is represented as a circle split in four parts, related to the tasks, the locations, the services and the contacts. The user "rotates" the lens so that the current perspective, i.e., the focus of interest, is in the top position. A selection of an instance of a task, location, service or contact reveals in the other quadrants an iconic view of the information items linked to the instance. A central section defines the temporal zoom of the lens, from day to year, whose selection coherently updates the four quadrants' content.

The second screenshot shows the lens focused on an instance of a task; the number of icons in the other quadrants matches the number of instances of locations, services and contacts for that task. Icon variants describe different categories of instances (e.g., contact groups and service types). Locations are

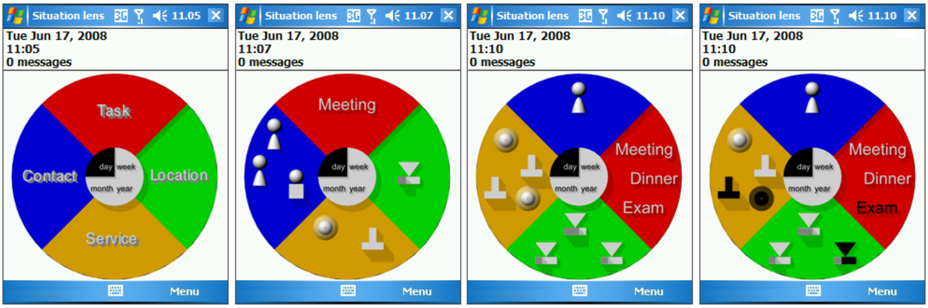



**Fig. 4.** The situation lens global perspective interface

marked with different levels of grey to show the distance from the user location, sensed by a position sensor or assumed as a static information from the PIM database.

Tapping and holding on an icon shows identification labels (names) and a menu for information filtering and direct access to the detail view. Tapping on an icon changes the situation lens focus to the icon. The third screenshot shows the lens focused on a person, displaying the tasks, locations and services linked to that person. A menu item associated to each icon allows the user to relate instances of the entities, as shown in the fourth screenshot. Keeping the situation lens focus on the person, the selection of a specific task highlights the subset of locations and services displayed on the screen that are associated to that task.

## 6 Conclusion

We have implemented a partial prototype in the Windows Mobile Device Simulator environment of Microsoft Visual Studio, to verify the feasibility of the approach both from the data management and from the user interaction points of view.

The implementation is at an early stage: data access and services interconnection have been tested, but the global perspective interface is a stand-alone module, allowing the user to explore the relations among the entities by "rotating" the lens, but acting on a set of data simulated with internal data structures.

The use of the simulator on as desktop PC lessens also the perception of the interface usability, due to the different gestures which distinguish mouse based and pen based operations. Also, the new generation of multitouch devices like the Apple iPod and iPhone will open new possibilities for natural gesturing. Nevertheless, the early experiments show that switching between entities improves the user perception of the relations among the tasks, mainly for activities long lasting or frequent, often bound to a common base of social and professional relations.

Compared to the situated metaphor of Hewagamage and Hirakawa [5], our approach is less complete but more flexible and more oriented to situation facets linked to the user perception and goal rather than physical status. Hence, the diminished possibilities of planning and automatically executing services concern only the observable parameters of the user status. Our approach gives more importance to the decisions the user takes according to a personal and subjective evaluation of the objective information at hand.

Further work will be devoted to implement and to integrate the situation lens interface on Windows Mobile systems and to test it in a number of professional environments, starting from the academic and research worlds whose features match the scenario we have described.

## References


1. Bertelsen, O., Bodker, S.: Activity theory. In: Carroll, J. (ed.) HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science, pp. 291–324. Morgan Kaufmann, San Francisco (2003)
2. Clancey, W.J.: Situated cognition: on human knowledge and computer representations. Cambridge University Press, Cambridge (1997)
3. Suchman, L.: Plans and Situated Actions. Cambridge University Press, New York (1987)
4. Dourish, P.: A foundational framework for situated computing. Position paper for the CHI 2000 Workshop on Situated Computing: A Research Agenda (2000)
5. Hewagamage, K.P., Hirakawa, M.: Situated computing: A paradigm to enhance the mobile user's interaction. In: Chang, S.K. (ed.) Handbook of Software Engineering and Knowledge Engineering. World Scientific, Singapore (2000)
6. Bardram, J.: Plans as situated action: An activity theory approach to workflow systems. In: ECSCW (1997)
7. Bolchini, C., Quintarelli, E.: Filtering mobile data by means of context: A methodology. In: Proc. OTM Workshops 2006 - Context-Aware Mobile Systems, pp. 1986–1995 (2006)
8. Bolchini, C., Schreiber, F.A., Tanca, L.: A methodology for very small database design. Information Systems 32, 61–82 (2007)
9. Tanca, L.: Context-based data tailoring for mobile users. In: Proc. BTW Workshops 2007, pp. 282–295 (2007)
10. Berhe, G., Brunie, L., Pierson, J.M.: Distributed content adaptation for pervasive systems. In: ITCC 2005, International Symposium on Information Technology: Coding and Computing, Las Vegas, Nevada, USA, pp. 234–241 (2005)
11. Costa, P.D., Pires, L.F., van Sinderen, M.: Architectural patterns for context-aware services platforms. In: Proc. of the 2nd International Workshop on Ubiquitous Computing, IWUC 2005, Miami, FL, pp. 3–18 (2005)
12. Strang, T., Linnhoff-Popien, C.: Service interoperability on context level in ubiquitous computing environments. In: Intl. Conf. on Advances in Infrastructure for Electronic Business, Education, Science, Medicine, and Mobile Technologies on the Internet, L'Aquila, Italy (2003)
13. Bederson, B.B., Clamage, A., Czerwinski, M.P., Robertson, G.G.: DateLens: A fisheye calendar interface for PDAs. ACM Trans. Comput.-Hum. Interact. 11, 90–119 (2004)
14. Karlson, A., Bederson, B., SanGiovanni, J.: AppLens and LaunchTile: Two designs for one-handed thumb use on small devices. In: Proc. of SIGCHI Conference on Human Factors in Computing Systems, pp. 201–210 (2005)

# A System for Dynamically Generating User Centric Interfaces for Mobile Applications and Services

Abayomi Ipadeola, Oludayo Olugbara, Matthew Adigun, and Sibusiso Xulu

Center of Excellence for Mobile e-Services
Department of Computer Science, University of Zululand, South Africa
profyommexy@yahoo.ca, oluolugbara@gmail.com,
madigun@pan.uzulu.ac.za, ssxulu@pan.uzulu.ac.za

**Abstract.** Device heterogeneity, diversity in user preferences and uncertainty of a steady network bandwidth allocation are challenges, which must be squarely addressed to achieve effective information communication in a mobile computing environment. Browser-based user interface adaptation has been widely adopted as a feasible solution to these challenges. Moreover, with the introduction of J2ME platform, some inherent challenges of the browser-based environment for information communication have been surmounted. This paper describes a system for generating user centric interfaces for mobile applications and services. A user centric interface considers the needs and the preferences of the user. We regard a user interface component as an artifact and user interface generation involves dynamic selection of suitable artifacts based on real-time preferences of the user. The system has proved to be efficient for an on-demand generation of user centric interfaces and it provides support for direct user participation during interface generation. Apart from generating user centric interfaces, the system achieves a high degree of dynamism, flexibility and supports design time testing and evaluation of the generated interfaces.

**Keywords:** User Interface, Multi-Device Interface, Interface Adaptation, Interface Generation.

## 1 Introduction

The recent advancement in wireless technology, which brought about the emergence of different mobile networks, such as 2G, 3G and 4G [1] has significantly enhanced information communication. This advancement has led to the design of portable devices, popularly called Personal Communication Devices (PCD) [2]. PCD are heterogeneous and are of varying characteristics. Mobile networks and PCD can be described as the main agents for the realization of ubiquitous computing environment. In spite of this advancement, there are still some inherent challenges confronting efficient information communication in a typical mobile computing environment. These challenges largely arose from the characteristics of a personal communication device. Typically, PCD are personal, portable and heterogeneous. The heterogeneity of PCD, diversity in user preferences and uniqueness of execution environment demand that software applications accessed by PCD are adapted to these unique attributes for successful information communication [3].

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 175–184, 2008.

The objective of this paper is to describe an authoring system based on polymorphic logical description for generating user centric interfaces for consumers of mobile services. The system, called CoMADE achieves direct user participation in user interface generation and supports modularity, allowing mobile service developers to work simultaneously on polymorphic artifacts. The rest of this paper is summarized as follows. Section 2 reviews the related work in multi-device user interface design. Section 3 gives high level description of the system and the important steps used by the system for interface generation. Section 4 considers system implementation details and sample application description. Section 5 gives a brief conclusion.

## 2 Related Work

The focus in user interface adaptation research was initially directed towards achieving generic models that can be automatically adapted for diverse devices without having to create separate interfaces for every device. A hybrid of patterns and model based approach [4] and Quality of Information Mark-up Language (QIML) [5] were then proposed for realizing multi-device interfaces. In [6], the use of multiple logical descriptions for achieving multi-device interfaces with three levels of abstraction was proposed. The work in [7] proposed the use of multi-client collaborative approach for addressing small screen web browsing difficulty. The approach involves two levels browsing scheme for within page and between page comparative browsing. A Multi-Device Authoring Technology (MDAT) was developed in [8] for designing form-based web applications that can execute on heterogeneous devices. All these approaches however addressed the multi-device capability of user interfaces without a support for user participation in user interface adaptation. Moreover, many of the existing approaches targeted user interfaces for browser-based environments without a support for J2ME platform, which provides a multi-device capability for addressing the diversity in computing devices. The support for user participation for achieving user centric interface generation is however not considered in these approaches. The need for user interface generation through direct user participation is obvious, due to pervasive integration of PCD into our operating environment. As a result, an Intent Oriented Approach (IOA) for multi-device user interface generation was proposed [9]. Intent oriented based approaches, however, limit user participation to the selection of media content, such as images for an interface presentation and the diversity of user preferences are not considered during user interface generation.

We describe a system for automatic generation of user centric interfaces for individual consumers of mobile applications and services. The system is based on the concept of polymorphic logic description, which combines polymorphic task decomposition [10] and conventional model based approach [11] to enable user participation in the process of interface generation. We achieved direct user participation in interface generation for J2ME enabled devices. The J2ME platform offers some benefits, which are unavailable in a browser-based environment. These benefits include among others, proficient security system, disconnected access and synchronization provisioning, cross-platform compatibility, dynamic delivery of applications and services, enhanced user experience, scalability and performance [12]. Additionally, in our approach, user participation is not limited to selection of media content, but users can directly influence the selection of every interface artifact during an on-demand user interface generation.

## 3 The CoMADE System Architecture

Fig 1 shows the system architecture of CoMADE, which comprises of two core components, namely, Polymorphic Logical Descriptor (PLD) and Mediator. The PLD contains Polymorphic Task Model generator (PTMg), Polymorphic Abstract Model generator (PAMg) and Polymorphic Concrete Model generator (PCMg) as sub-components. Similarly, the mediator contains Content Adaptation Engine (CAE), User Interface Generator (UIG). The User Information Repository (UIR), Interface Object Repository (IOR), Media Content Repository (MCR) and Request Information Bus (RIB) are other components that play important roles in the generation of user centric interfaces.

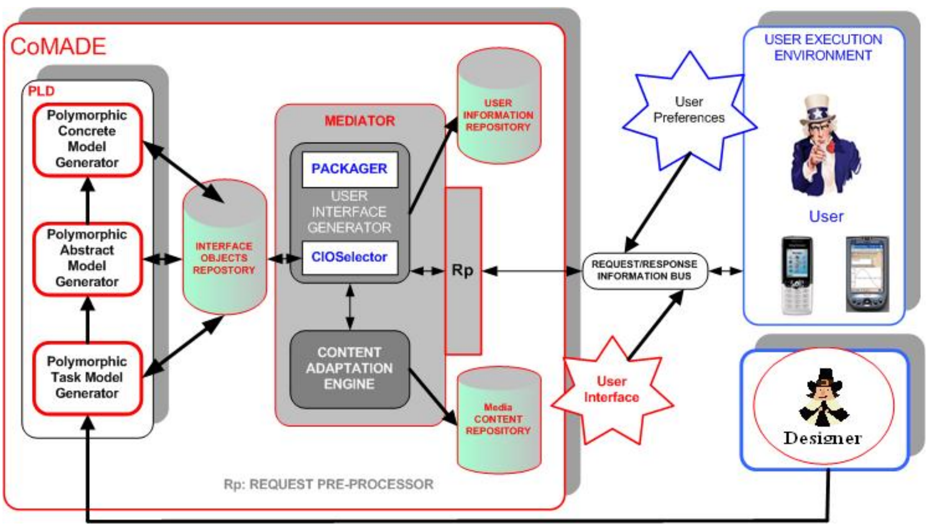



**Fig. 1.** CoMADE System Architecture

### 3.1 The Polymorphic Logical Descriptor

Polymorphic logical description is a user interface description specified at design time for addressing the diversity of user needs, requirements or preferences based on the conventional model based approach for generating user interfaces and based on this approach, PLD is realized. Fig. 2 shows the detail description of the operational principles of CoMADE system. First, an interface designer specifies tasks and associated variants. Tasks are the logical activities that make up a user interface and are performed in order to reach the user's goal [13]. Variants are task alternatives, which are introduced to cater for the diversity in user preferences.

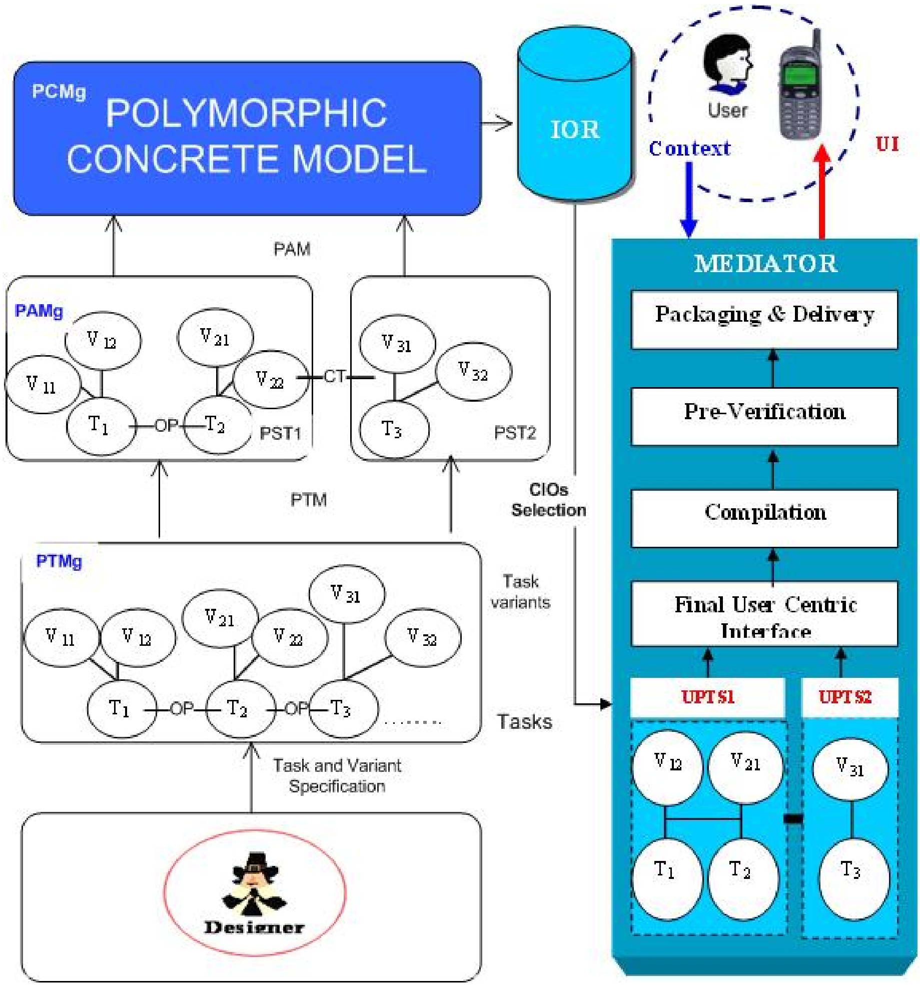



**Fig. 2.** Steps for User Centric Interface Generation with CoMADE

The PTMg, which is based on polymorphic task decomposition [10] is concerned with the creation of a knowledge base of Task Variant Information (TVI). The knowledge base gives the descriptions of the various tasks to be performed by a user on an interface. It also contains information about task variants associated with each defined task. A variant is referred to as an alternative representation of a task. Variants are introduced to handle the diversity in user preferences.

The PAMg, relies entirely on the output of the PTMg for the realization of a device-independent description of user centric interface. The task variants represented in a knowledge base are then converted into Abstract Interaction Objects (AIO) using information associated with each task. For instance, an object such as “EditInput” is generated for an input task, such as “Enter Your PassCode”. An AIO therefore, is realized for every variant associated with a given task. In addition to the generation of

AIOs for each variant, the PAMg organizes the task-variant information into Presentation Task Sets (PTS) for an interface. A PTS represents tasks that are enabled over the same period of time [6]. It is sometimes referred to as presentation unit or a window perceivable by a user at a given time [14]. A PTS consists of a set of tasks and each task is associated with a number of variants. A collection of PTS and its associated connections make up the polymorphic abstract model, which is an important input to the PCMg.

The PCMg is concerned with the realization of a polymorphic concrete interface, which is a device-dependent description of an interface. The PCMg converts AIO variant objects in each PTS into Concrete Interaction Objects (CIO). This involves the generation of platform implementation representation for every AIO variant object in a PTS. For instance, a concrete description for an EditInput object is realized at this stage. For the target platform, an implementation representation for "textfield" object, which is a CIO object, is generated for the "EditInput" object. Such representation is persisted into a knowledge base. For every PTS, a concrete description is generated for its variant abstract interaction objects of the PAM. The concrete description generated by the PCMg is stored into an object database, which is represented in the Interface Object Repository (IOR).

### 3.2 The Mediator

The mediator is an important component that generates the final user centric interface for a requesting user. The sub-components of the mediator are the User Interface Generator (UIG) and the Content Adaptation Engine (CAE). The sub-components of the UIG include Packager and the CIOSelector. The Media Content Repository (MCR) of CoMADE stores various contents for user interface presentation. It can be referred to as an interface resource repository. Media contents, such as images, video and audio files are managed by this component. During an interface generation for a requesting user, an appropriate content is selected and adapted by the CAE. A Request pre-Processor (Rp) component interprets user requests received from the request/response information bus before they are dispatched to the mediator.

The CAE handles the adaptation of particular media content and it relies on user preference information as well as the profile of a requesting device for the selection and adaptation of a media file. Images can be grayed or sampled. An audio content can be substituted with a video content. Also an adapted video content can be projected to users based on retrieved preferences. User preferences are retrieved either at design time, through the aid of an explicit user model or at runtime through an extended proxy device.

The User Interface Generator (UIG) is an important component of the mediator as it carries out the generation of the final user centric interface. It utilizes the information from other components to generate a user centric interface for a requesting user. When a request for a user interface is received by the mediator, the CIOSelector is activated for the selection of CIO variants from every PTS. The selection is based on pre-recorded user preferences or real-time preferences received through an explicit proxy device. The CIOSelector enables at most a CIO variant (task variants in CIO form) to represent a task in every task node in a PTS. The set of CIO variants selected in a PTS makes a User centered PTS (UPTS). The UIG creates a class template for every interface. A class template is a MIDLet file generated for a set of UPTS.

The packager processes the generated user centric interface for final delivery. This processing involves compilation of all needed files, pre-verification of class files, creation of manifest, packaging into Java Archive File (JAR), creation of Java Application Descriptor file (JAD) and the generation of a Universal Resource Location (URL) that points to the generated user-centric interface.

## 4 Implementation of an Authoring Environment

Fig. 3 shows a snapshot of CoMADE authoring environment, which was implemented for generating user centric interface based on PLD. The system has capability for polymorphic task, abstract and concrete user interface modeling. Polymorphic task and abstract models are represented in Task Variant Description Language (TVDL) and Presentation Set Description Language (PSDL) respectively. TVDL and PSDL are modeling languages based on eXtensible Mark-up Languages (XML) for interface


**Fig. 3.** Snapshot of the CoMADE System

description. Polymorphic concrete model contains concrete interaction object representation for each abstract interaction object in PSDL. For a J2ME platform, CoMADE achieves concrete representation by the generation of Java-implementation description for each abstract interaction object in the PSDL. For instance, the J2ME representation for an EditBox is generated and persisted into an object database. DB4o was selected due to its known advantages [15]. During design time, interface designers can update polymorphic concrete description and enable CIOs with additional functionalities. CoMADE achieved user centric interface through dynamic selection of CIO based on user requests. MIDlet classes are generated and packaged together with user interface descriptor files for individual users. Moreover, a mediator, example-preference test section and emulator programs are integrated with CoMADE for design time testing of generated interface. With CoMADE, a designer can easily extend an existing user interface simply by specifying additional variants to cater for one or more user preferences. The problem of direct user participation that we set to accomplish was solved by CoMADE through dynamic selection of interface artifact and individual user interface generation based on user preference.

An application was developed for an on going Grid-based Utility Infrastructure for SMME Enabling Technology (GUISET) project [16]. In GUISET environment, service consumers subscribe to different services and are provided with user interface based on their preferences for service subscriptions. Figs. 4 and 5 show a scenario involving two service consumers $C_1$ and $C_2$ with different preferences. Fig. 4 shows a service consumer $C_1$ that subscribes to Business News (LN), Stock Quote and Sport News (SN). Similarly, Fig. 5 shows a service consumer $C_2$ that subscribes to Word for the Day, Stock Quotes and Jokes. The diversity in the services requested by these two consumers requires that different interface presentations be generated for these consumers. The generated interfaces should therefore, suit the consumers according to their preferences.


**Fig. 4.** Interface for Consumer 1


**Fig. 5.** Interface for Consumer 2

An experiment was conducted to evaluate the effectiveness of the system in generating user interfaces according to specified user preferences and its user friendliness. Sample interface applications were generated by volunteered students of the university whose level of expertise in interface design varies considerably. An initial questionnaire was administered for categorizing students into expert, novice and intermediate levels. Fig. 6 shows that 43%, 54%, and 63% of expert, intermediate and novice strongly agreed that the system generates user interfaces according to the specified user preferences. Similarly, 57%, 38% and 25% of expert, intermediate and novice agreed that CoMADE generates user interfaces according to the specified user preferences, but 8% and 12% of intermediate and novice could not decide. Fig. 7 shows that 100%, 31% and 50% of expert, intermediate and novice claimed that CoMADE is very easy to use. Similarly, 31% and 25% of the intermediate and novice claimed that CoMADE is easy to use, but 38% and 25% of intermediate and novice could not decide.

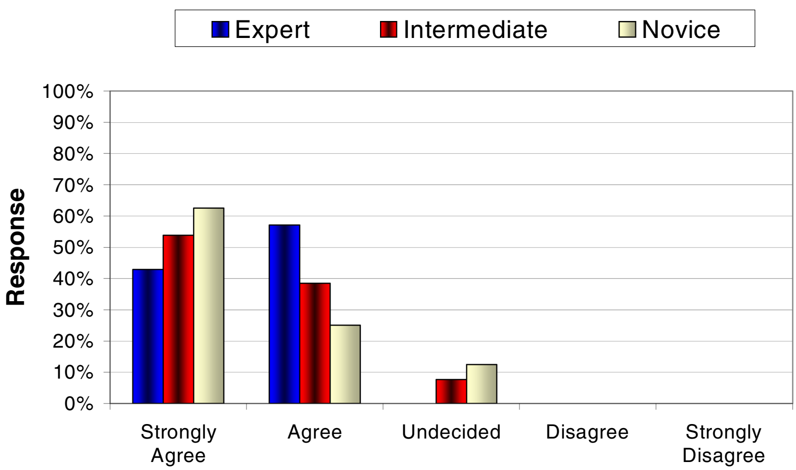



**Fig. 6.** Generated Interfaces Met Specified Preferences

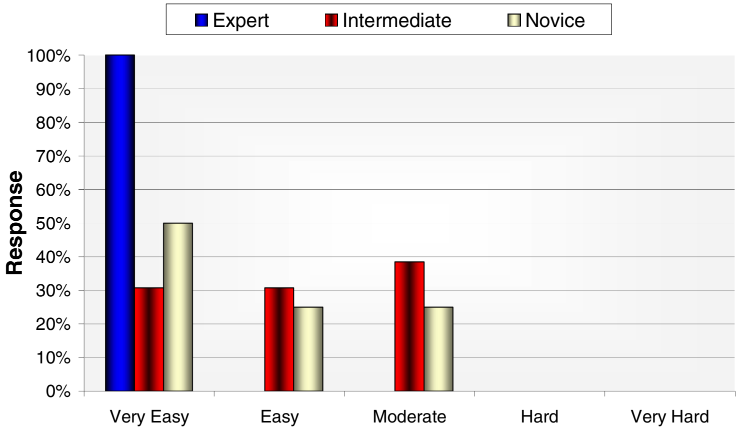



**Fig. 7.** Response to Ease of Use

## 5 Conclusion

A separate design of user interfaces for every device has proved to be inefficient due to its gross disadvantages. In order to achieve universal usability of interfaces, systems support for automatic user centric interface generation becomes important. We describe an authoring system for the generation of user centric interfaces for consumers of mobile applications and services. The system allows designers to simultaneously work on polymorphic artifacts. An interface is generated on-demand based on pre-recorded user information or real time preferences. Direct user participation, ease of application extensibility and high degree of flexibility and dynamism are achieved.

## References

1. Yuan, M., Long, J.: Impact of 3G and 4G wireless networks on Higher Education, `http://www.cce.iisc.ernet.in/iche07/41.pdf`
2. Barbara, b.: Designing Mobile User Experience. John Wiley & Sons, Chichester (2007)
3. Estrin, D., Culler, D., Pister, K., Sukhatme, G.: Connecting the Physical World with Pervasive Network. In: IEEE Pervasive Computing, January- March 2002, pp. 59–69 (2002)
4. Lin, J.: Damask: A Tool for Early-Stage Design and Prototyping of Cross-Device User Interfaces. In: Conference Supplement of UIST 2003: ACM Symposium on User Interface Software and Technology, Vancouver, BC, pp. 13–16 (2003)
5. Huang, C.M., Chao, Y.C.: Universal WWW Access for Heterogeneous Client Devices. In: Euromicro Conference, p. 0315 (2001)
6. Mori, G., Paterno, F., Santoro, C.: Design and Development of Multi-device User Interface through Multiple Logical Descriptions. Proceedings on Software Engineering 30(8) (2004)
7. Zhigang, H., Hanqing, L.: Web Browsing on Small-Screen Devices: A Multiclient Collaborative Approach. IEEE Pervasive Computing 5(2), 78–84 (2006)
8. Guruduth, B., Lawrence, B., Richard, C., Vianney, C., Yves, G., Frederique, G., Christine, H., Shin-ichi, H., Masahiro, H., Fumihiko, K., Goh, K., Ashish, K., Ono, K., Schade, A., Danny, S., Winz, K.: An Authoring Technology for Multi-device Web Applications. IEEE Pervasive Computing 3(3), 83–93 (2004)

9. Bisignano, M., Di-Modica, G., Tomarchio, O.: An Intent-oriented Approach for Multi-Device User Interface Design. In: Proceeding of the 20th International Conference (April 2006), http://ieeexplore.ieee.org/xpl/RecentCon.jsp?punumber=10777
10. Savidis, A., Paramythis, A., Akoumianakis, D., Stephanidis, C.: Designing user-adapted interfaces: the unified design method for transformable interactions. In: Proceedings of the conference on designing interactive systems: processes, practices, and techniques (1997)
11. Paterno, F.: Model-based tools for pervasive usability. Interacting with Computers 17(3), 291–315 (2005)
12. Integrated Java Technology for End-to-End m-Commerce, http://developers.sun.com/mobility/midp/article/mcommerce/
13. Paterno, F.: Multimodality and Multi-device interfaces. In: W3C Workshop on Multimodal Interaction, Sophia Antipolis, France (2005)
14. Szekely, P.: Declarative Interface Models for User Interface Construction Tools: The Mastermind approach.: Engineering for Human-Computer interaction. In: Bass, L., Unger, C. (eds.), pp. 120–150. Chapman& Hall, London (1995)
15. Adigun, M.: Zululand University: On Our Way to Wireless Grid Computing. Presentation at University of Syracuse,USA (2006)
16. Grehan R.: Complex Object Structures, Persistence, and db4o. Open Source Object Database in Java and .NET, http://www.db4o.com/about/productinformation/whitepapers/

# Multimodal Mobile Virtual Blackboard

Danco Davcev[1], Vladimir Trajkovik[1], and Sladjana Gligorovska[2]

[1] University “Ss Cyril and Methodius”, Skopje, Macedonia,
Faculty of Electrical Engineering and Information Technologies
www.feit.edu.mk
[2] GISDATA Skopje, Macedonia
www.gisdata.com.mk

**Abstract.** Our multimodal mobile virtual blackboard system enables different forms of communication: chat, VoIP, draw, file exchange. Making small screens of mobile devices more usable can be done by adapting features that an application offers according to the specific user preferences and to the current utilization of the application. In this paper, we describe our approach in development of a multimodal mobile virtual blackboard by using XML agents and fuzzy logic. The general opinion among participans for the interface usability is positive.

**Keywords:** Mobile Learning, Multimodal systems, XML Agents, Fuzzy logic, Pocket PC.

## 1 Introduction

Development of mobile communication devices and standards in wireless communication and networks has enabled people to use the devices in many different ways. One of the most honorable utilizations of this new mobile technology is mobile learning [1]. Mobile learning is a new way of acquiring knowledge, which is highly adaptable [2], [3] to different kinds of student profiles [4], from people that do not have time to attend normal courses to a practical enhancement of ordinary courses with additional access to the knowledge. Besides the static contents that are provided in books and various digital multimedia [5], very important aspect of learning is exchange of knowledge and consultation process among student participants and available expert authority (professor/instructor) [6].

There are many desktop applications that support multimedia communication among participants. Communication among several participants which includes multimedia transfer is at the beginning of the development and utilization in the sphere of mobile handheld devices. Mobile handheld devices' primary use was personal assistance, making schedules and notes, and voice communication. New demands are imposed to these devices: greater processing power to support real time multimedia transfer (video stream, voice stream, file exchange), greater memory space and data transfer. Modern, feature rich applications demand space on the screen for displaying all available features, which could be an issue when handheld devices are used.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 185–194, 2008.

A great consideration should be given to the readability and utilization of small mobile user interfaces. Further step towards achieving usable interfaces on small screen is layout adaptation of utilized features in the application. We have developed a multimodal user interface that automatically can switch from one view to another according to the specific user preferences and the current usage of the specific media. Our multimodal knowledge – based interface is managed by three XML agents and fuzzy logic.

The related work is presented in the second section of this paper. Mobile blackboard architecture and design are elaborated in the third section. The fourth section describes the management of the agent-based user interface. The Mobile Blackboard implementation and evaluation follow in the fifth section. Finally, we conclude the paper.

## 2 Related Work

In [1] learner-centered handheld tools were presented. We have designed similar tools, but we have extended their usability. They can be used for more diverse cases, since they support uploading of any material as bitmap picture and allow drawing and making notes. Creations of new sketches are also possible, along with the immediate consultation with other students and the instructor.

In [2] an intelligent tutoring system was presented. It is different from our system because we have introduced static agents whose purpose is to intelligently adapt the user interface to the preferences of each user.

In [3] an adaptation of the user interface based on the device used in the learning process was elaborated. In our case, we have developed context aware learning system which determines the features of the user interface according to the learning contents for the specific learning session. It can be also used to dynamically reconfigure the user interface according to the user preferences and learning contents.

The system that stimulates students' active involvement in learning process by using the manual personalization of PDA devices and organization of folders is elaborated in [4]. In our approach, we propose agent based adaptive personalization in the learning process.

Instead of using hardware-based personalization [6] we try to achieve similar effects by applying adaptive interfaces on the standard devices.

The agent-based approach that uses fuzzy logic to determine importance of certain information is elaborated in [7]. We use similar approach to determine importance of user interface features in regards to user preferences and media contents used in current session.

Effect of Input Mode on Inactivity and Interaction Times of Multimodal Systems [8] is another research that pays attention to the experience of the users in using variable offered modes of interaction with the user interface. We also have made tests to understand the affinity of the usage of our application's features.

## 3 Mobile Blackboard Architecture and Design

User interface for mobile devices should have two characteristics:

1. Capacity to show the learning contents in an appropriate way (clear, understandable, with appropriate size);
2. The user interface should be adaptable to the user.

Our mobile virtual blackboard system enables different forms of communication: chat, VoIP, draw, file exchange. A user or a whole consultation group can focus on some of the possible features, for example, VoIP and draw, ignoring the chat utility. In the case of handheld devices, where screen surface is a scarce resource, a feature that is not used should be invisible in order to relinquish the space to a more frequently utilized feature. That is why the necessity of multimodal user interfaces is obvious. For the management of the user interface, three different agents have been incorporated into our virtual blackboard consultation system. These agents monitor user interaction with the available features of the application and the media contents in the particular consultation session. They also update the user profile and calculate the visualization features of the user interface.

In this way, we propose highly adaptable multimodal user interfaces for handheld devices which can bring benefits to application developers for mobile software environment, to users of mobile learning systems and finally, to all mobile device users.

There are two kinds of clients in our Virtual Mobile Blackboard application, both implemented on a pocket PC platform. The first client is instructor application, which controls the resources in the group, and is the source of relevant knowledge. The other client is student application dedicated to use the knowledge provided from the instructor along with mutual communication and consulting. There is only one instructor and many students in one consultation session. Network communication is realized in a way that clients send UDP datagrams to the server, and the server resends the UDP datagrams, according to the contents of the received message. For voice transfer, streams are used. Desktop based clients (for both professor and students) can be also connected to the server.

The client user's interface has controls for chat, draw, VoIP, file upload/download, authority control (on use of the blackboard and VoIP channel).

The instructor can give the control over the blackboard (or by giving the speaking possibility) and reclaim the control later. Instructor has the role of a moderator in the consultation process.

Interactivity of this system is very important feature. Technology limitations should not interfere with the student-professor interaction and discourage the users of the system. Many messengers available today, have accustomed people to use this kind of communication and collaboration. Implementing consultations on pocket PCs provides convenience by making people more available. Another important point is that the student's focus of interest (blackboard, where the topic is being explained) is at the same place where the student takes notes, or draws own scenario to show to the rest of the group. Traditionally, the professor draws on table/paper, or explains certain topic on a projected fixed picture on the wall and students watch and take notices in their notebooks. Merging these two attention focuses, the virtual table consultation system provides prompt discussions and greater attention. Speaking, combined with drawing, where also the students largely participate, should be the most appreciated way of conducting conversation and explaining learning topics.

Fig. 1 shows the design architecture of the consultation system. Client applications reside on pocket PCs and they have three logical parts: Active user agent, VoIP client and the client application. Active user agent manages user interaction within the application regarding the interface visibility and utilization of the features. VoIP client

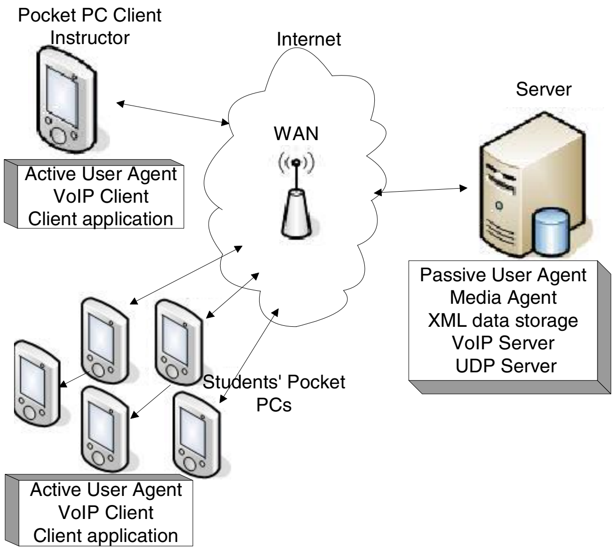



**Fig. 1.** Architecture of the mobile blackboard system

serves the voice transfer, and the client application represents all the features mentioned above (chat, draw, voice, file up/download, communication among clients). Both kinds of clients (student and professor) have the same architecture. The communication is realized through wireless area networks connected to the Internet. The server contains Passive User Agent, Media Agent, XML data storage, VoIP Server and UDP server application. For the purpose of saving the memory and processing power of the clients handheld devices, Passive User Agents and Media Agent reside on the server side. The agents are XML based, so the data storage is also XML based.

## 4 Multimodal Agent-Based User Interface Management

Readability and good usability of the screen is of a great importance for handheld devices and demands great developer attention, since the success of the application depends on the users' positive experience using the application interface. Every particular user can have his/her own preferences for the utilization of the features in the application, and the consultation session can vary, according to those preferences and the material that is to be presented. Regarding the user interface dimensions on handhelds, a management of the interface is needed, in order to expose only the currently used features. We have created multimodal user interface that exposes only these currently used features. In this paper, we describe our approach in development of a knowledge-based user interface by using XML agents and fuzzy logic.

The Active user agent resides on the client side, and monitors the user interaction with the features in the application. At the start of the consultation session, it communicates with the Passive user agent on the server. The Passive agent retrieves information from the Media agent, about the type of the materials to be used for the particular consultation session. Media agent monitors the folder with the materials for the consultation session, and sends the information to the Passive agent. Passive agent then calculates the user interface according to the info from Media Agent, history of interface affinities for the particular user and Active agent message on use of the features and sends the corresponding information (for the type of the user interface) to the Active agent which adapts the user interface according to this information.

User profiles with personal information for every user that can be logged are stored in the XML database Additionally, a history of affinities is stored for every user. According to the history, Passive agent adapts the user interface on the beginning of the consultation session for each user.

The part of the XML structure of the Passive User Agent is shown above. It is composed of two parts: Affinities, which contains the data about user's affinity, and Rules, according to which, the passive agent decides the type of the user interface.

There are three predefined user interfaces, which can be preloaded on the client's device, according to his preferences and the media contents for the current consultation session. The first interface has chat on the larger part of the screen, the second one has larger drawing blackboard on the screen, and the third one has equal space for the chat and the blackboard. VoIP feature doesn't occupy much space on the screen, so the buttons for voice are always displayed.

Since there is no universal conclusion for making decisions of which user interface should be preloaded, i.e. which interface mode should be used, the fuzzy logic approach is used to model that kind of the imprecise information [7].

In order to create fuzzy expert system for calculation of the user interface visual features, fuzzy variables are defined for this expert system. The linguistic variable LEVEL accepts values from the set of terms {high, medium, low} as shown on Figure 2. It can be used to represent chat, draw and voice affinity.

The interaction made by the user on the features in the application (chat, draw and voice feature), is represented as a vector (chat_aff, draw_aff, sound_aff). A linguistic variable named *RESULT_PREFERENCE* is introduced, and it accepts values from the set of terms {increase_chat, increase _draw (= reduce_chat)}. This variable represents

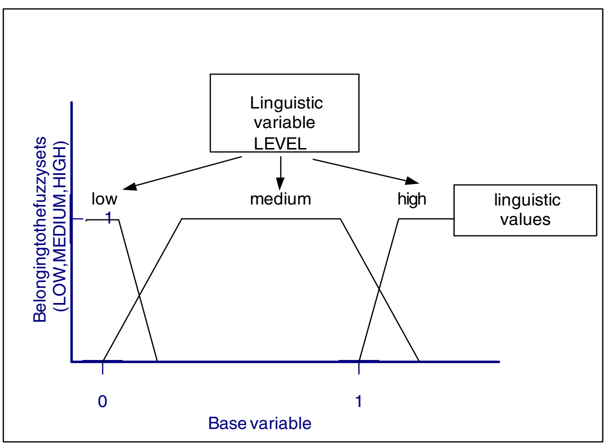



**Fig. 2.** Graphic representation of the defined fuzzy functions

the visualization of the user interface, produced by the user interaction with the interface and with the contents of media folder.

The proposed expert system has two variables: U_A (user affinity) which presents the most used feature by the user, i.e. it has the greatest affinity for the feature; M_P (media profile) presents the profile of the interface that should be used according to the media contents. The output variable named I (interface) presents the necessary interface to be preloaded onto the user's device screen, according to the input variables.

The system contains Active agents on each client handheld machine and one Passive User Agent and also a Media Agent on the server machine. Media Agent monitors the specific folder with the materials for the consultation session. Passive User Agent serves all the Active User Agents in one consultation session. Interaction among the three clients is shown on the Figure 3. In this class diagram, used attributes along with the methods are shown for each agent.

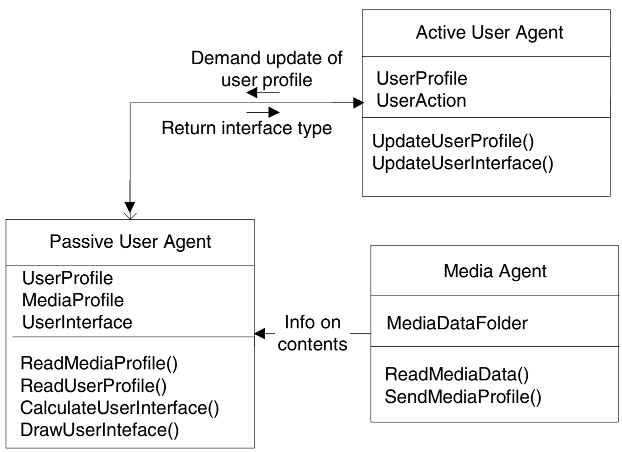



**Fig. 3.** Agents' model

Passive User Agent uses the following attributes: UserProfile, MediaProfile and UserInterface. MediaProfile is variable which contains the information of the prepared media contents for this consultation session. Regarding the contents of the folder where the files for upload are stored, MediaProfile can be text, drawing, or sound. The necessity of introduction of this variable can be shown with simple example: user might not prefer to draw, but the explanations are done with the help of the drawings. Therefore, drawing surface should be shown on the screen. Passive User Agent's main task is to calculate the appropriate user interface, according to user's affinity and contents of the media for the current consultation session. Active User Agent monitors user's utilization of the media features (responds to user's interaction on the interface: whether it uses the draw function, or chat or voice). After the user action, Active User Agent updates the user's profile and triggers the Passive User Agent to reconsider the user's interface type with the updated data. The interaction among agents and the user interface is shown on the Figure 4.

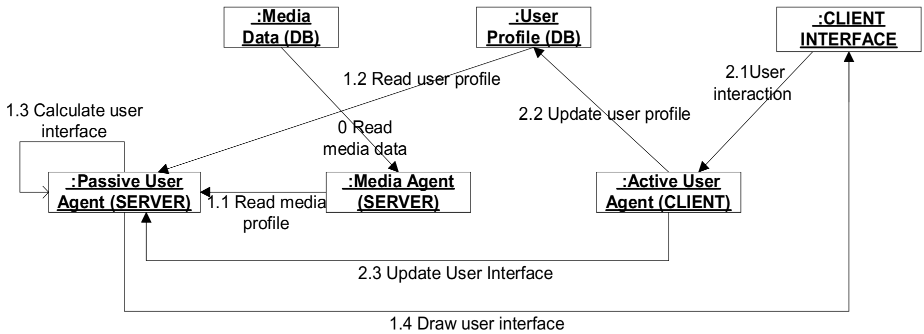



**Fig. 4.** Agents' collaboration diagram

**Table 3.** Communication example**.** A simple scenario for exchange of messages among the agents.

| Active Agent | Passive Agent | Media Agent |
|---|---|---|
| <message ID = "0" SessionID = "434" recipient ="passive_agent017" > < userID> Peter </ userID > < action >start_app</ action > </message> | < message ID = "1" SessionID = "434" recipient ="media_agent001"> </message> <!--processing info with passive_agent XML code--> < message ID = "4" SessionID = "434" recipient ="active_agent017"> < userID> Peter </ userID > <interface> draw</ interface > </message> | < message ID = "3" SessionID = "434" recipient = "passive_agent001" > <media_contents>draw</media_contents > </message> |

The multimodal structure of the user interface is made with the help of these three autonomous agents that serve only for user interface purposes.

## 5 Mobile Blackboard Implementation and Evaluation

Applications for Wireless mobile virtual blackboard are first made in C++ Visual Studio .NET 2005 development environment. Operating system used on the pocket PCs was Windows Mobile 2003. The application is upgraded to the Visual Studio 2008, and the Pocket PC operating system is changed to Windows Mobile 5. Operating system for desktop machines on which the application is practically deployed, is Windows XP SP2.

The two instances of instructor's interface are shown on Figure 5. The control of the blackboard is represented with the button named "get control". The chat module consists of the input text box as well as display text box, where all chat messages are written, and the send button. The button named "Chat" enables seeing the chat on the full screen. The buttons named "fUp" and "fDown" serve for upload and download.

The place where the graphics is shown is called the virtual blackboard. This is the place where the downloaded files are opened and where notes can be added using colors and pens with different thickness. The “Clean” button serves for clearing the blackboard. With clicking on the “Save” button, the contents shown on the virtual blackboard can be saved as a bitmap file.

Student's user interface is very similar to the professor's except that it has a button to send drawing and voice request; it does not have the authority control, and does not have the ability for upload.


**Fig. 5.** Two instances of mobile blackboard interface

Figure 5 shows two types of user interfaces, managed and adjusted by the user interface management system with three presented agents.

The mobile blackboard was evaluated by 24 students at their last year of undergraduate studies in Information Technologies. The students were asked to assess whether the mobile blackboard would be useful for their studies. The questionnaire given to the students included questions regarding interface usability. General opinion among participans for the interface usability is positive (86%). Interface is functional and suggestions for interface rearrangment are mostly done according to the user needs (76%).

The questions that examine interface usability and provide information for future upgrades are:

1) Are the colors from the drawing pens enough?
2) Are you satisfied with the way of presentation on the blackboard?
3) What do you think of giving drawing controls to more than one user?
4) What is your opinion on the logging concept on the system?

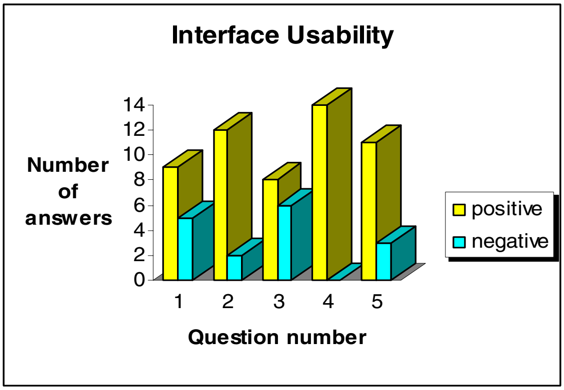



**Fig. 6.** Results from interface usability

5) What is your opinion on the controls position on the user interface and do you have any suggestions about it?

The X axis in Figure 6 represents questions with the same number from questionnaire above.

General opinion among participans for the interface usability is positive. Interface is functional and suggestions for controls position will be included in the future project. Option for more than one users to have drawing controls divided participants. The current system design left this option for the instructor who can decide about number of drawing control holders.

## 6 Conclusion

This paper presents our approach that uses agents and fuzzy logic in order to achieve multimodal Mobile Learning Pocket PC. We have developed context aware learning system which determines the features of the user interface according to the learning contents for the specific learning session. It can be also used to dynamically reconfigure the user interface according to the user preferences and learning contents. Fuzzy logic approach is used to model user affinities to different features of the interface. In this way, we proposed a highly adaptable user interface for handheld devices which brings benefits to the developers of applications for this software environment, to users of mobile learning systems and at the end, to all mobile device users. General opinion among participans for the interface usability is positive.

## References

1. Luchini, K., Quintana, C., Soloway, E.: Design Guidelines for Learner-Centered Handheld Tools. In: CHI 2004, Vienna, Austria, vol. 6(1), pp. 135–142 (2004)
2. Glavinic, V., Rosic, M., Zelic, M.: Agents in m-Learning Systems Based on Intelligent Tutoring. In: Stephanidis, C. (ed.) HCI 2007. LNCS, vol. 4556, pp. 578–587. Springer, Heidelberg (2007)

3. Jeong, C., Lee, E.: Context Aware Human Computer Interaction for Ubiquitous Learning. In: Smith, M.J., Salvendy, G. (eds.) HCII 2007. LNCS, vol. 4558, pp. 364–373. Springer, Heidelberg (2007)
4. Mifsud, L., Morch, A.I.: That's my PDA!' The Role of Personalization for Handhelds in the Classroom. In: Proc. of the Fifth IEEE International Conference on Pervasive Computing and Communications Workshops, pp. 187–192 (2007)
5. Gang, Z., Zongkai, Y.: Learning Resource Adaptation and Delivery Framework for Mobile Learning. In: Proc. 35th Annual Conference, Frontiers in Education, 2005. FIE 2005, October 19-22, 2005, pp. F1H-18- F1H-24 (2005)
6. Chang, A., Ishii, H.: Personalized Interactions with Reality Based Interfaces. In: CHI 2006 Workshop What is the Next Generation of Human-Computer Interaction? (2006)
7. Camacho, D., Hernandez, C., Molina, J.M.: Information Classification Using Fuzzy Knowledge Based Agents, vol. 4, pp. 2575–2580 (2001)
8. Perakakis, M., Potamianos, A.: The Effect of Input Mode on Inactivity and Interaction Times of Multimodal Systems. In: Proceedings of the 9th international conference on Multimodal interfaces, pp. 102–109. ACM, New York (2007)

# Personalized Mobile Multimodal Services: CHAT Project Experiences

Giovanni Frattini, Federico Ceccarini, Fabio Corvino, Ivano De Furio, Francesco Gaudino, Pierpaolo Petriccione, Roberto Russo, Vladimiro Scotto di Carlo, and Gianluca Supino

ENGINEERING.IT S.p.A. ,Via Antiniana 2/A, 80078, Pozzuoli (NA), Italy
{giovanni.frattini,federico.ceccarini, fabio.corvino,ivano.defurio, francesco.gaudino,pierpaolo.petriccione,roberto.russo, vladimiro.scottodicarlo,gianluca.supino}@eng.it

**Abstract.** Despite optimistic expectations, the spread of multimodal mobile applications is proceeding slowly. Nevertheless the power of new high-end devices gives the opportunity to create a new class of application with advanced synergic multimodal features. In this paper we present the results the CHAT group achieved in defining and building a platform for developing synergic mobile multimodal services. CHAT is a project co-funded by Italian Ministry of Research, aimed at providing multimodal context-sensitive services to mobile users. Our architecture is based on the following key concepts: thin client approach, modular client interface, asynchronous content push, distributed recognition, natural language processing, speech driven semantic fusion. The core of the system is based on a mix of web and telecommunication technologies. This choice proved to be very useful to create high personalized context sensitive services. One of the main features is the possibility to push appropriate contents on the user terminal reducing unfriendly user interactions.

## 1 Introduction

While mobile devices are becoming increasingly powerful and sophisticated, the need for new and complex services has amplified. Anyway, complex services could loose usability, thus new generation mobile applications must be built adopting convenient inputs and outputs modes in order to make dialog natural and intuitive and, as much as possible, user centric.

One of the possible ways to improve usability is multimodality.

Advances in single mode interaction, like multitouch, are having good success, but synergic multimodal applications remain uncommon and in some context, e.g. mobile devices, they are practically absent. This cannot be attributed to hardware capabilities as the most part of the latest PDAs and smartphones is enough powerful to guarantee a satisfactory multimodal/multimedia experience to final users. So we expect that in a short time, multimodal mobile applications will gain the scene. The aim of CHAT project is designing and developing a platform for building multimodal services dedicated to cultural heritage fruition (museums, archeological parks) and e-learning. The

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 195–207, 2008.

ambition is to offer synergic multimodality in a mobile context using common devices. We have concentrated our attention on synergic multimodality (simultaneous use of different modalities in a single interaction act), considering that alternate multimodality (a multimodal sequence of unimodal messages) has been investigated during past years [1].

We have focused our research on the following main topics:

- a client-server approach where business logic, output contents, unimodal input recognizers and fusion are all located on the server side;
- Distributed Recognition (Speech, Sketch, Handwriting);
- development of smart ways to collect inputs from unimodal channels, analyze and merge them (fusion) to get an interpretation of the whole interaction act;
- definition of a language to build multimodal interfaces (XUL-like approach);
- asynchronous push of contents toward mobile devices;
- speech recognition based on natural language processing.

The input channels we considered are: “audio” (for speech input); “point” (for the act of pointing an object on the screen); “ink” (for freehand stylus interaction, sketch/handwrite; “context” (e.g. geographical localization); “keyboard”.

## 2 Related Works

Commercial products focused on multimodality are still uncommon. Kirusa [www.kirusa.com] is a relatively small company that produces multimodal software for telecom operators. Kirusa core product is a solution for Voice SMS, which is a relatively simple multimodal application. Nuance [www.nuance.com], a well known brand in the speech recognition, has recently released a framework for building multimodal mobile applications. This framework is based on standards like X+V and, thus, limited to alternate multimodality. It is worth noting that Nuance has chosen a distributed recognition approach limiting the processing on client side.

In [1] [2] we discuss and highlight some of the topics that, in the meantime, the CHAT project has faced.

It is interesting that, while for alternate multimodality, several mark-up languages have been proposed and adopted industrially (namely X+V) [3], for synergic multimodality does not exist a standard language, even if several researches have addressed the problem proposing possible solutions [4]. We studied how to extend existing languages (X+V, SALT, SMIL[5]) to multimodality, but the final decision was to implement our lightweight language called LIDIM. This decision comes also from other experiences. For example, MONA project [6] has produced a multimodal presentation server, which main features is to support deploy of device independent applications combining GUI and speech input/output. We have found particularly interesting the MONA capabilities to push pages on mobile devices using UIML, a markup language for building user interfaces. MONA is based on a client side engine that allows the presentation of a non standard mark-up language and the push of pages on the mobile device (for low-end phone, WAP push messages). In CHAT we use the IP signaling protocol (SIP) to send unsolicited and trigger multimodal data retrieval.

Thinlet framework [7] has inspired us in finding a solution for aggregating multimodal objects in a single UI while the idea of generating UI on the fly is discussed in

[8]. In [2] we have discussed an embryonic prototype of a J2ME mobile multimodal interface. For the .NET implementation of such interface we have extended the Piccolo framework [9], an open source library for graph visualization.

For speech recognition we have used a commercial product by Loquendo [10]. For handwriting recognition we have adopted the open source solution Jarnal [11]. For sketch recognition we have used software based on algorithms by IRPPS [12] [13].

Simultaneous use of complementary communication channels, leads to a more complicated input processing. Voice is probably the best option to communicate while eyes and hands are busy and it represents the best mean to express complex questions. The next generation of voice based user interface technology enables easy-to-use automation of new and existing communication services, achieving a more natural human-machine interaction. The main trouble in managing human-machine dialog lies in bridging the gap between machine's jargon and natural language. As reported in [14], for a determined task, some linguistic events are essential to recognize and understand, others not so. In a user's utterance not all words have got the same semantic weight and research aims are more and more related with information extraction or part of speech clustering and classification, as shown in [15][16]. Main of the current works on information extraction and dialogue understanding systems are still based on domain-specific frame-and-slot templates. Anyway, this strong link between quality of recognition and restricted application context is the real limit to overcome. So in [17] is proposed a shallow semantic interpreter based on semantic roles that are slightly domain-specific. This work describes an algorithm to identify semantic roles in a sentence. Such algorithm leverages on statistical techniques that have been successful for the related problems of syntactic parsing, part of speech tagging, and word sense disambiguation, including probabilistic parsing and statistical classification.

To build natural language processing systems, the most followed empirical approach starts from a training corpus comprising sentences paired with appropriate translations into formal queries. Then statistical methods are used to face the semantic parsing problem [18]. Typically statistical algorithms are trained on a hand-labeled dataset; one of the most accurate is the FrameNet DB [19].

## 3 Overall System Architecture

The image below shows the system architecture.

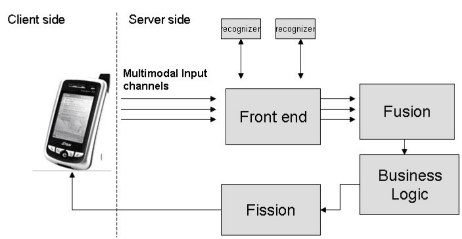



**Fig. 1.** Overall system architecture

Such architecture comes from well known patterns [20]. The main innovation we introduced concerns the usage of a mix of web 2.0 and telecommunication technologies, this has produced interesting results. Nevertheless, this approach has different impacts on how software on terminals has to be designed.

For the client side we follow a "thin client approach". So client side there is just an application able to show a good looking graphical interface, to manage the output channels and to collect the inputs from the user and context information. We call such application MMUIManager (MultiModal User Interface Manager) and it is able to use net protocols (HTTP/RTP) to talk with other blocks of the architecture.

The Front End block contains the interfaces between the MMUIManager and the logic of the architecture. Such interfaces are mainly servlets. Such block has also the task to collect the inputs coming from unimodal channels, send them to recognizers (speech, sketch, handwriting). After the recognition, if successful, each interface generates an EMMA [www.w3.org/TR/emma] output that is sent toward a fusion module and then forwarded to business logic.

The Fusion module merges contemporary inputs coming from different channels. It receives different inputs in EMMA format and merges them in a single EMMA file.

The Business Logic block has two main tasks to accomplish: maintain/adjourn the status of the interaction for each user and select the contents to send back according to request, user profile and context.

The Fission module has to distribute the contents selected by Business Logic to final users. Such contents are distributed in a specific language called LIDIM.

## 4 A Flexible Client Interface

MMUIManager follows a "thin client" approach. So it is a light client which has to accomplish the following tasks:

- show a good looking interface built on the fly according to the indication given from the server side business logic;
- collect multimodal input and show multimedia output to have a good system interaction and content fruition experience.

To reach such goals we have followed a XUL like approach [21]. So we have defined a specific language (LIDIM) to describe a multimodal/multimedia mobile interface and we have included in MMUIManager a browser for such a language that builds the multimodal interface and shows it to final user. Such approach allows us to have a very flexible interface that can be built dynamically according to physical context, interaction type, user profile. The graphical aspect, but the contents too, do not reside in the device, but they are dynamically sent by business logic.

We need a modular interface which is able to collect multimodal inputs (speech, sketch, handwrite, point, context) and to send them toward front end server, able to manage multimedia output (text, image, audio, video) that can be composed on the fly. So we have developed a framework made of a set of elements that can be composed to build a coherent multimodal interface. Elements available are buttons, textboxes, html browsers, imageboxes, audio/video panels (which are elements related to audio/video resource that can be played to final user) etc. Each element can be enriched with specific input modalities, like touch or draw. So you can have an image box, sensible to stylus, you can draw on; such an element can acquire freehand inputs (sketch and/or handwrite).

Ink traces are saved in InkML format and, after the acquisition is ended, they are sent toward the front end server.

Audio acquisition is done in a totally different way. It is not related to a specific object of the interface but is captured by the MMUIManager itself; so it is done continuously and what is acquired is sent via RTP directly to speech recognizer.

The XUL-like approach we have followed makes us need a language able to describe modular, multimodal/multimedia interface  A first possibility is to adapt an existing language to our demands that are: synergic coordinated multimodality, multimedia output and modular interface. But this proved to be difficult: SMIL has been created for output and it is not simple to adapt it to input, while X+V and SALT have not been created for synergic multimodality or to manage different input channels. So we decided to create our own language. The result of our efforts is LIDIM (in Italian LInguaggio Definizione Interfaccia Multimodale, e.g. language for designing multimodal interface).

LIDIM manages the following output channels:

- graphic modular panel interface;
- audio (pre-recorded sources and TTS);
- Text Messages (Pop up message).

For each resource of the modular interface it can be specified one or more output media (image, audio, video, text) and, eventually, one or more input modality (point, sketch). Note that LIDIM has only the goal to describe the output interface, while MMUIManager has the aim to build the interface and manage input/output channels.

We give a sample of how it is possible to build an object having as output media text, image and audio resource and as input mode point and sketch.

```
<object id="78678687">
    <output_media>
        <text uri="http://...."/>
        <image uri=" http://...."/>
        <audio uri=" http://...."/>
    </output_media>
    <input_mode>
        <point isActive="true"/>
        <sketch isActive="true"/>
    </input_mode>
</object>
```

Like HTML, LIDIM does not include the resources (images, text, audio/video resources) which compose the interface; it just contains the addresses where MMUIManager can get them.

The language is template based, so it leaves to MMUIManager to dispose objects on the screen. Another limitation is the impossibility to synchronize different output resources. Future releases could exceed such limitations.

We have built our MMUIManager prototype with different technologies. Each platform has its strengths and weaknesses. At the moment we have developed prototypes of such multimodal framework in J2ME and .NET Compact Framework. Such choice comes from two different and someway opposite demands:

- the possibility to run on almost every commercial devices;
- the necessity to exploit all the hardware and OS capabilities of the devices.

.NET implementation started from an existing framework: the Carnegie Mellon Piccolo[9]. Starting from Piccolo objects we have created multimodal objects, eventually linked in a graph, able to collect inputs and manage outputs.

Up to now .NET implementation is far better as it is able to manage streaming for audio input and audio output while for J2ME we weren't able to do the same as no standard support for RTP protocol is provided. So the support for continuous speech recognition is not present in J2ME version and speech channel is managed via HTTP.


**Fig. 2.** Example of client interface

## 5 The Front End Module, the Approach to Distributed Recognition

In our architecture we have chosen to distribute the recognition process outside the device to have lighter client application as mobile devices have limited hardware resources. With such an approach, it was suitable to have an abstraction layer among MMUIManager and recognizers. Such layer is the Front end module that essentially has the following tasks to accomplish:

- redirect input from the MMUIManager toward the recognizers, if necessary;
- send the recognized inputs toward the fusion module in EMMA format.

For Distributed Speech Recognition, as we want to recognize speech in real time, there must be an open streaming channel (RTP protocol) between the device and the recognizer (ASR, Automatic Speech Recognizer). This allows the continuous recognition of the user speech.

To open the channel there must be a negotiation process between MMUIManager and recognizer; such negotiations are made using the MRCP protocol.

For Distributed Sketch/Handwriting Recognition, there's no need of a real time process. After the MMUIManager has acquired an InkML traces set, it sends it via POST HTTP to the Front end module which forward it to recognizer.

Essentially the Front-End has been implemented as a group of cooperating Java Servlets deployed on an application server (e.g. Tomcat, JBoss etc.).

The figure below shows the overall organization of the module.

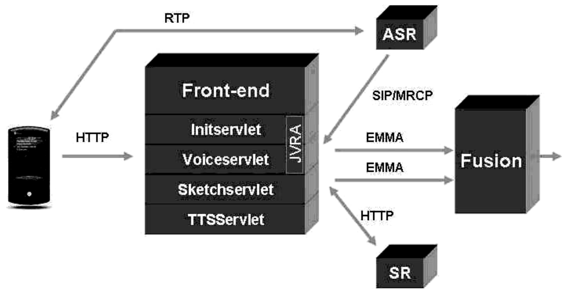



**Fig. 3.** Architecture of Front End module

The servlets we implemented are the following ones:

- **InitServlet:** instantiates a SIP communication channel between the ASR and the client device. A specific method returns to the client the IP of the ASR, the port and the MRCP port in order to open the voice channel.
- **SpeechServlet:** recognizes the output of the ASR and builds EMMA file to sent to the Fusion.
- **SketchServlet:** builds an EMMA file with the symbol ID and others additional parameters.

The mechanism for instantiation of the voice channel and initialize voice recognition has been implemented as follows:

1. Client invokes the InitServlet in order to open the RTP channel;
2. InitServlet contacts the ASR and returns to the client the specified MRCP port to open a direct RTP channel to the ASR;
3. The client device invokes the ASR and opens SIP channel;
4. The ASR tries to recognize each voice input from the device. If a sentence is recognized, the result with a confidence value is sent to the servlet.
5. The VoiceServlet analyzes the output of the ASR and checks the associate confidence value. If this value is higher than a default value, an EMMA file with the recognized sentence is built. Otherwise, if the confidence value is lower than the default, a compensation process starts with an action in order to specify to the client an error message using TTSServlet.
6. A complete EMMA with the recognized sentence is sent to the Fusion module.

**Java Voice Resources Adapter (JVRA)**

We developed a specific module, JVRA, to manage distributed automatic speech recognition (ASR) and text-to-speech (TTS) systems. JVRA implements MRCP V2.

Media Resource Control Protocol Version 2 is a protocol being developed by the Speech Services Control Working Group (Speechsc) of the Internet Engineering Task Force (IETF) [https://www.ietf.org/mailman/listinfo/speechsc] to allow client hosts to control media processing resources such as speech synthesizers, speech recognizers, speaker verifiers and speaker identifiers residing in servers on the network.

MRCP commands between the client and the server are used to establish (bi-directional) audio paths between the client and server using the Real time Transport Protocol (RTP). RTP packets sent from the client application to the MRCP server contain the caller's speech, commonly referred to as utterances, sent for recognition. Conversely, RTP packets sent from the server to the client contain either TTS (speech synthesis) or streamed audio retrieved from a web server.

RTP traffic is encoded using the G.711 codec [Telecommunication Standardization Sector (ITU-T) - http://www.itu.int/ITU-T] to ensure quality audio for recognition and delivery to the caller.

JVRA provides support for a subset of the ASR, TTS and generic commands defined within the MRCP specification. In particular, JVRA supports the following events and commands:

| **MRCPv2 Resource Type: speechrecog** | |
|---|---|
| **Methods** | **Events** |
| RECOGNIZE<br>DEFINE-GRAMMAR | START-OF-INPUT<br>RECOGNITION-COMPLETE<br>COMPLETE |
| **MRCPv2 Resource Type: speechsynth** | |
| **Methods** | **Events** |
| **SPEAK**<br>**STOP** | **SPEAK-COMPLETE**<br>**IN-PROGRESS**<br>**COMPLETE** |

The DEFINE-GRAMMAR method, from the client to the server, provides one or more grammars to speech recognizer. We used SRGS (Speech Recognition Grammar Specification [http://www.w3.org/]) to write this grammar. The RECOGNITION-COMPLETE is an event from the recognizer resource to the client indicating that the recognition completed. We used EMMA to describe recognition result.

The recognizers we are using are:

- ASR Loquendo [10] for speech;
- Jarnal for handwriting [11];
- Software based on algorithms developed by IRPPS (Istituto di Ricerche sulla Popolazione e le Politiche Sociali, Rome) for sketch recognition [12] [13].

## 6 The Approach to the Fusion

Fusion module has to accomplish the following tasks:

- to merge inputs from complementary channels into a single command;
- to signal and possibly solve ambiguous communication act from the user.

Simultaneous use of complementary communication channels, such as voice, keyboard and stylus (sketch/handwrite) leads to a more complicated input processing system, which has to take care of inputs temporary correlation, and user's habits.

Furthermore, the most appropriate contents rendering has to be chosen, on the base of the environment in which the user is acting, his preferences and device's capabilities.

Communicating by voice is probably the best option when eyes and hands are busy. Speech is also the most appropriate and instinctive mean to express complex questions to a service, especially when the information content of user's requests becomes more and more rich and complicated. Users often need to say something that is not exactly the answer to a question from the system. They may want to express their goals with complex sentences that answer more than one question at a time.

The consequent processing of natural language has to face the issue of extracting the original semantic content from the utterance. The idea underlying our approach is that for a given task in a particular context, some linguistic events are crucial to recognize and understand others not so. This way leads to ignore false starts, repetitions or hesitations typical of natural language, focusing on really significant speech fragments [14].

The meaning of linguistic utterances can be derived using formal structures, or meaning representations. The need for these representation models arises while trying to bridge the gap from linguistic inputs to the non-linguistic knowledge of the world needed to perform tasks involving the meaning of linguistic inputs.

Simply accessing the phonological, morphological, and syntactic representations of sentences may be not enough in order to accomplish needed tasks. For example, answering questions requires background knowledge about the topic, about the way questions are usually asked and how they are usually answered. Therefore the use of domain-specific knowledge is required to correctly interpret natural language inputs. An empirical approach to construct natural language processing systems starts from a training corpus comprising sentences paired with appropriate translations into formal queries. Then statistical methods are used to face the semantic parsing problem [18].

Here we propose a solution for a multimodal speech driven architecture.

The process of semantic inputs disambiguation is based on the collaboration of a stochastic classifier and a rule based engine. A domain specific data corpus has been manually labelled to build the model language while a named entity recogniser works as classifier, in order to extract meaningful semantic contents. User commands are rebuilt on server side through a frame filling process [22] in which the frame structure is application dependent. Frames are schematic representations of situations involving various participants, props, and other conceptual roles, each of which is a frame element [19] [17]. In this frame based approach, inputs coming from different modalities are converted in interconnected structures [23][24].

A Hidden Markov Model is adopted for modelling user utterances; hidden states of HMM are frame's semantic slots labels, while observed words are the slot filling values [25]. The aim of classifier is to detect the most likely category for a word (or cluster) given the overall sequence of previous words in the sentence.

The task solving process is carried out integrating information fragments included in frame's slots with the ones from complementary input channels. Available data are then processed by a rule engine reasoning using inference rules in a forward chain. In building our architecture we are aware about a crucial issue to overcome: the lack of information due to channel loudness [26]. We are studying how recover contents affected by such problem. A possible recovery algorithm is based on the Levin and Rappaport Hovav linking theory [27][17] that argues that the syntactic realization of predicate arguments is predictable from semantics. Therefore we expect that, fixing the semantic of user's utterance, it is possible to gather indications on the syntactic collocation of words helping in retrieval of corrupted information fragments.

## 7 The Approach to the Fission

For the Fission module we have implemented an asynchronous output system. We want the business logic to have the possibility to modify the interface shown to final users even without an explicit request from the user himself. We want the server side to be able to "push" content toward devices. Such a feature could be very interesting to have a real adaptive application, so that the user interface could be modified even without an explicit request from the user. To implement such a feature we have had to study a particular architecture for the module of Fission.

The multimodal fission component implements appropriate logical distribution of information into one or more outputs in different modalities. The task of a Fission module is composed of three categories [28]:

- *Content selection and structuring*: the presented content must be selected and arranged into an overall structure;
- *Modality selection*: the optimal output modalities are determined according to the current situation of the environment;
- *Output coordination*: the output on each of the channels should be coordinated so that the resulting output forms a coherent presentation.

According to the architecture of our system, the tasks of Fission process are linked directly to the business logic layer. The latter has the responsibility to provide an input to the Fission module that contains information taking into account factors such as user's preferences and behavioural characteristics. The Fission module, instead, must be able to send information contents in a coordinated manner: for example, it is responsible for guarantying synchronization between the visual presentation and the audio reproduction of the content. This is done creating a LIDIM file to send to MMUIManager. LIDIM file contains information about contents to visualize, interaction modality and resource synchronization.

The Fission module has been designed so that the architecture works in asynchronous way: when the business logic has a new content for a user, it signals to the Fission (HTTP). Then the Fission module notifies the new content to the MMUIManager module through SIP protocol. Then the MMUIManager module receives from the Fission the new content through a HTTP request.

SIP is the IETF's standard for multimedia conferencing over IP. SIP is an ASCII-based, application-layer control protocol (RFC2543) that can be used to establish, maintain, and terminate calls between two or more end points.

The *Dialog Interface* is the module through which the business logic layer talks to the Fission module for the notification of new contents. The *Push Manager* is responsible for the notification to MMUIManager of the presence of new content. The *Content Interface* is the component through which the MMUIManager module communicates with the Fission module to retrieve new contents.

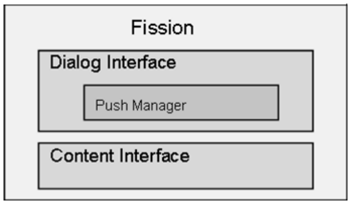



**Fig. 4.** Fission module internal architecture

As mentioned earlier, contents notification to the MMUIManager is managed through SIP protocol. The Push Manager achieves this notification, after a registration process to the SIP server. To complete contents notification, the PushManager performs a SIP call to MMUIManager which hangs up and gets ready to receive the contents. The figure below shows the steps described. Notice that the SIP server is not a part of Fission module, but it is a component of the overall system.

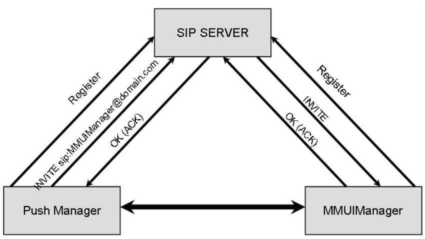



**Fig. 5.** Asynchronous push

## 8 Conclusion and Future Works

In this paper we have shown that a platform for developing mobile synergic multimodal application can exploit a mix of technologies that, together, could yield to interesting results. Among the others, the possibility to push contents on mobile handsets, using personalized criteria could add real value. We are aware that pushing contents could be perceived as a kind of spam and, thus, it must be used considering the user explicit wills.

There are several open issues and research domain that have been just partially explored. We are absolutely aware that speech recognition in noisy environments is still not robust enough. This is why new techniques for improving speech understanding and eventually recovering the sense of a phrase from additional inputs and/or from context information could lead to interesting results. This is a research domain that could be considered as a part of a larger effort to improve fusion mechanisms for making mobile multimodal applications viable. Furthermore, the usage of sketch or handwriting as input channels in distributed environment is not common: recognizers are generally located client side and an effort to understand how to use them in a distributed environment could contribute to their diffusion.

## References


1. Frattini, G., Romano, L., di Carlo, V.S., Petriccione, P., Supino, G., Leone, G., Autiero, C.: Multimodal Architectures: Issues and Experiences. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM 2006 Workshops. LNCS, vol. 4277, pp. 974–983. Springer, Heidelberg (2006)
2. Frattini, G., Gaudino, F., di Carlo, V.S.: Mobile multimodal applications on mass-market devices: experiences - 18th International Workshop on Database and Expert Systems Applications (DEXA 2007) Dexa workshop 2007, pp. 89–93 (2007)
3. X+V for the Next Generation Web, http://www.voicexml.org/specs/multimodal/
4. Filippo, F., Krebs, A., Marsic, I.I.: A framework for rapid development of multimodal interfaces. In: Proceedings of the 5th international conference on Multimodal interfaces, Vancouver, British Columbia, Canada, November 05-07 (2003)
5. Synchronized Multimedia Integration Language (SMIL), http://www.w3.org/TR/REC-smil/
6. Niklfeld, G., Anegg, H., Pucher, M., Schatz, R., Simon, R., Wegscheider, F., Gassner, A., Jank, M., Pospischil, G.: Device independent mobile multimodal user interfaces with the MONA Multimodal Presentation Server. In: Proceedings of Eurescom Summit 2005, Heidelberg, Germany, April 27-29 (2005)
7. Thinlet classic (last accessed March 20, 2007), http://www.thinlet.com/
8. Reitter, D., Panttaja, E., Cummins, F.: UI on the fly: Generating a multimodal user interface. In: Proceedings of Human Language Technology conf 2004 / North American chapter of the Association for Computational Linguistics (HLT/NAACL 2004) (2004)
9. Piccolo (last accessed February 20, 2008), http://www.cs.umd.edu/hcil/jazz/index.shtml
10. Loquendo vocal technology and services, http://www.loquendo.com
11. Jarnal (last accessed March 5, 2008), http://www.dklevine.co/general/software/tc1000/jarnal.htm
12. Avola, D., Ferri, F., Grifoni, P.: Formalizing Recognition of Sketching Styles in Human Centered Systems. In: Apolloni, B., Howlett, R.J., Jain, L. (eds.) KES 2007, Part II. LNCS (LNAI), vol. 4693, pp. 369–376. Springer, Heidelberg (2007)
13. Avola, D., Caschera, M.C., Ferri, F., Grifoni, P.: Ambiguities in Sketch-Based Interfaces. In: Proceedings of the Hawaii International Conference on System Sciences (HICSS 2007), p. 290. IEEE Computer Society, Los Alamitos (2007)
14. Gorin, A.L., Alonso, T., Abella, A., Riccardi, G., Wright, J.H.: Semantic Information Processing of Spoken Language - How May I Help You? In: Pattern recognition in Speech and Language Processing. CRC Press, Boca Raton (2003)
15. Wright, J.H., Gorin, A.L., Riccardi, G.: Automatic Acquisition of Salient Grammar Fragments for Call-Type Classification. In: Proc. 5th Europ. Conf. Speech Communication and Technology, pp. 1419–1422. Intern. Speech Communication Ass., Bonn, Germany (1997)
16. Arai, K., Wright, J.H., Riccardi, G., Gorin, A.: Grammar fragment acquisition using syntactic and semantic clustering. Speech Communication 27 (January 1999)
17. Gildea, D., Jurasfky, D.: Automatic labeling of semantic roles. Computational Linguistic 28(3), 496–530 (2002)
18. Ng, H.T., Zelle, J.M.: Corpus-Based Approaches to Semantic Interpretation in NLP. AI Magazine 18(4), 45–64 (1997)
19. Johnson, C.R., Fillmore, C.J.: The FrameNet tagset for frame-semantic and syntactic coding of predicate-argument structure. In: Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics (ANLP-NAACL 2000), April 29-May 4, 2000, pp. 56–62 (2000)
20. W3C multimodal interaction activity, http://www.w3.org/2002/mmi

21. XUL project (last accessed November 12, 2007), http://www.mozilla.org/projects/xul/
22. Minker, W., Bennacef, S.: Speech and Human-Machine Dialog. Kluwer Academic Publishers, Boston (2004)
23. Koons, D.B., Sparrell, C.J., Thorisson, K.R.: Integrating simultaneous input from speech, gaze and hand gestures. In: Maybury, M. (ed.) Intelligent Multimedia Interfaces, Menlo Park, CA, pp. 257–276. MIT, Cambridge (1993)
24. Russ, G., Sallans, B., Hareter, H.: Semantic Based Information Fusion in a Multimodal Interface. In: CSREA HCI (2005)
25. Jurafsky, D., Martin, J.H.: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition
26. Hakkani-Tür, D., Tur, G., Riccardi, G., Kim, H.K.: Error Prediction in Spoken Dialog: From Signal-to-Noise Ratio to Semantic Confidence Scores. In: The Proceedings of ICASSP-2005, IEEE International Conference on Acoustics, Speech, and Signal Processing, Philadelphia, USA (March 2005)
27. Levin, B., Rappaport Hovav, M.: Lexical Semantics and Syntactic Structure. In: Lappin, S. (ed.) The Handbook of Contemporary Semantic Theory, pp. 487–507. Blackwell, Oxford
28. Foster, M.E.: State of the art review: Multimodal fission (2002), http://www.hcrc.ed.ac.uk/comic/documents/deliverables/Del6-1.pdfM

# A General-Purpose Context Modeling Architecture for Adaptive Mobile Services

Thomas Pederson[1], Carmelo Ardito[1], Paolo Bottoni[2], and Maria Francesca Costabile[1]

[1] Dipartimento di Informatica, Università degli Studi di Bari, 70125 Bari, Italy
{pederson,ardito,costabile}@di.uniba.it
[2] Dipartimento di Informatica, Università di Roma La Sapienza, Rome, Italy
bottoni@di.uniroma1.it

**Abstract.** Mobile context-aware computing aims at providing services that are optimally adapted to the situation in which a given human actor is. An open problem is that not all mobile services need contextual information at the same level of abstraction, or care for all aspects of the user's situation. It is therefore impossible to create a unique context model that is useful and valid for all possible mobile services. In this paper we present a compromise: a three-tiered context modeling architecture that offers high-level mobile services a certain freedom in choosing what contextual parameters they are interested in, and on what abstraction level. We believe the proposal offers context modeling power to a wide range of high-level mobile services, thus eliminating the need for each service to maintain complete context models (which would result in severe modeling redundancy if many services run in parallell). Each mobile service must only maintain those parts of the context model that are application-dependent and specific to the mobile service in question. We exemplify the use of the context model by discussing its application to a mobile learning system.

**Keywords:** Context Model, Context Awareness, Mobile Services, Human-Computer Interaction.

## 1 Introduction

Because of the very nature of mobile devices, human interaction with them is strongly related to their context of use. Recent applications for mobile use try to take advantage of contextual information to offer better services to users. Pervasive, or ubiquitous, computing [5] calls for the deployment of a wide variety of smart devices and sensors throughout our working and living spaces, which not only can offer a more "intelligent" local environmental behaviour but also provide important contextual cues to mobile devices operating in the environment. The overall goal with these infrastructures combining wearable and instrumented computation power is to provide users with immediate access to relevant information and to transparently support them in their current tasks. As Human-Computer Interaction (HCI) systems expand beyond the virtual environment presented on a computer screen and start to encompass also real-world objects and places, the need to better conceptualize these new components of the system, as well as the intentions of the human agents currently operating the

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 208–217, 2008.

system, becomes pressing. Context-aware systems differ from traditional HCI systems not only because they utilize the state of the physical world as part of interaction, but also because they do it implicitly [4]. One might say that context-aware systems provide computational functionality directly or indirectly tied to real-world events without adding input devices but by gathering information in other ways (typically through sensors, of which the human is not necessarily aware).

The work presented in this paper is part of the CHAT project ("Cultural Heritage fruition & e-learning applications of new Advanced (multimodal) Technologies"), which aims at developing a general-purpose client-server infrastructure for multimodal situation-adaptive user assistance. In such architectures, dialogue management is typically based on the integration of independent components that execute specific tasks (sometimes, these components are "out-of-the-box", e.g. components for voice recognition). CHAT intends to develop and evaluate an architectural framework that facilitates implementation of such multimodal services.

In this paper, we use the term "context" as defined by Dey et al. [2]: "any information that can be used to characterize the situation of entities (i.e. whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves." An open problem is that not all mobile services need user-related contextual information at the same level of abstraction or care for all aspects of the user's situation. It is therefore impossible to create a unique context model that is useful and valid for all possible mobile services. In this paper we present a compromise: a three-tiered context modeling architecture that offers higher-level mobile services freedom in choosing what contextual parameters they are interested in, and on what abstraction level.

After a brief introduction to the Adaptive Dalogue Manager module, the rest of the paper focuses on the context model and its application to a mobile learning system.

## 2 The Adaptive Dialogue Manager

The framework proposed in the CHAT project aims at adapting the "dialogue" with the user according to several factors: the service provided, the task currently executed, the environment in which the user acts ("context"), the user him/herself and her/his device. These factors are measured and managed by a set of specific software components that together make up the the Adaptive Dialogue Manager as shown in Fig. 1. One of these software components is the Context Reasoner, pictured in the lower right corner, that creates and maintains the context model.

The Adaptive Dialogue Manager is the CHAT framework element in charge of a) identifying the most appropriate content to be returned to the client in order to satisfy user's request and b) determining the next system state by updating the models describing the different interaction factors. The Adaptive Dialogue Manager receives its input from a software module, called *Fusion*, which recognizes and combines low-level user input events from different channels (tap or sketch on the screen, voice, gesture, RFID or visual tag scan, etc.) in order to build an overall meaningful input. In a specular way, the output of the Adaptive Dialogue Manager, indicating the most suitable content to be delivered to the user on the basis of the overall interaction state, is refined by a *Fission* module. This retrieves or generates suitable forms of the (possibly multimedia-based) content and takes care of their delivery and synchronization aspects. In CHAT, the Adaptive Dialogue Manager runs on a server that communicates through wireless networks with mobile thin clients. However, nothing prevents (parts or the whole) of the manager to run locally on the mobile device in the future as the computing power of mobile devices increases.

As shown in Fig. 1, the Adaptive Dialogue Manager results from the interaction of different reasoners, each managing a specific model. In the next section we focus in particular on the context model, maintained by the Context Reasoner.

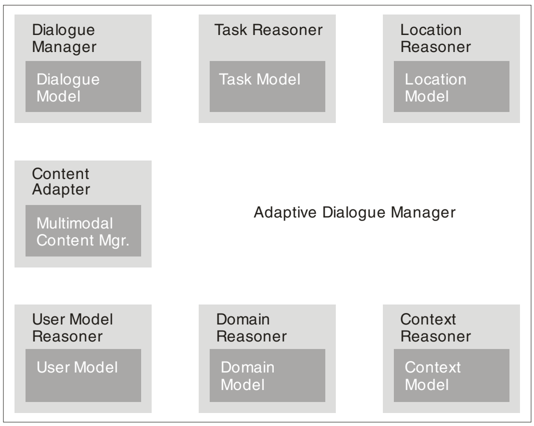



**Fig. 1.** The components of the Adaptive Dialogue Manager and their associated models

## 3 The Context Model

The context model is an aggregated body of information about a) environmental parameters that can be used to determine the user's current situation, and b) computation and interaction characteristics of the mobile device of the specific user.

In our view, an ideal context model should 1) provide information on context aspects relevant for the given service application; 2) hide irrelevant context details; 3) offer a high-level interpretation of lower-level context details if requested. The context model has to cope with different types of requirements. A low-level general-purpose context modeling (e.g. the identification of absolute geographical coordinates for a specific mobile device) has to serve both service applications and other Adaptive Dialogue Manager components (e.g. Dialogue Manager, Content Adapter, User Model Reasoner, Domain Reasoner, Task Reasoner) At a high, application-specific level, the system could exploit the previous information with respect to specific semantics (e.g. the fact that the user of the device has successfully passed all exhibition rooms in a specific museum and is heading for the exit). In order to deal with this variety of needs, we propose that the context model be distributed over three levels of abstraction, similar to [3], of which the Context Reasoner maintains the first two (low- and medium-level) as part of the Adaptive Dialogue Manager (see Fig. 2).

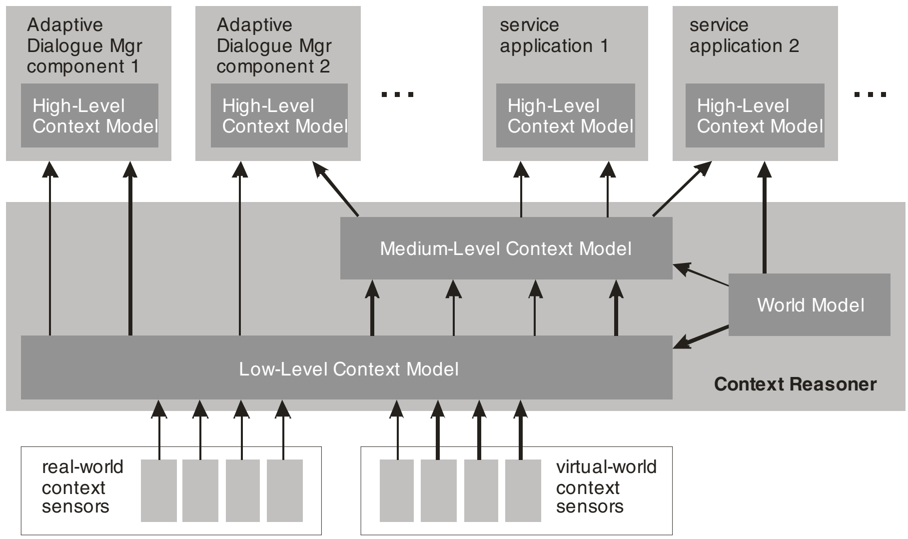



**Fig. 2.** A three-tiered context modeling architecture where the Context Reasoner maintains the two lower-level context models and leaves higher-level context modeling to other Adaptive Dialogue Manager components and service applications

"Real-world context sensors" are system entities providing information about real-world phenomena such as geographical position, temperature, etc. "Virtual-world context sensors" provide information about events happening inside computing devices such as cellular phones, as well as information about the devices' technical capabilities themselves. Higher-level models are maintained by components and services outside the Context Reasoner. Fig. 2 illustrates how high-level contextual information can be derived by service applications and Adaptive Dialogue Manager (ADM) components by selecting and combining attributes from the three resources made available by the Context Reasoner, namely:

1. the Low-Level Context Model;
2. the Medium-Level Context Model;
3. the World Model.

In Fig. 2, the ADM component 1 chooses to create and maintain its internal high-level context model on the basis of low-level context attributes only (e.g. the time of day, the GPS coordinates of a mobile device). ADM component 2 makes use of both low and medium-level context attributes (e.g. the time of the day and the semantic location of a mobile device). Service application 1 uses only medium-level context attributes, while service application 2 combines medium-level context attributes (e.g. geographical speed) with information from the World Model database (e.g. the maximum speed allowed on a specific road). Each of the three Context Reasoner components is described in more detail below.

**Low-Level Context Model**

The Low-Level Context Model (LLCM) is continously updated by the Context Reasoner with the status of available real-world context sensors as well as status information about the software environment running on the mobile device (denoted as “virtual-world context sensors” in Fig. 2). The functional requirement of the LLCM is to be able to capture all aspects of context which can be useful to improve context-aware performance of the specific service applications targeted within a given development project. In the case of the CHAT project we have decided to let the LLCM capture and provide the following low-level context properties:

- *time*
- *absolute position* of a device, given directly by sensor
- *relative position* (e.g. given by proximity to objects included in the world model carrying Bluetooth position transmitters or visual tags)
- *device capabilities* (e.g. CPU power, client-server bandwidth; battery, etc.)
- *status information of available sensors* (both wearable – e.g. accelerometers, electronic compass – and environment-based ones.)

**Medium-Level Context Model**

The Medium-Level Context Model (MLCM) incorporates a set of context attributes created by the Context Reasoner by combining low-level context attributes provided by the LLCM and information from the World Model according to rules which are universally applicable for all foreseeable services that the CHAT architecture is imagined to host. We propose the MLCM to contain the following set of attributes:

- *approximate absolute position* derived by combining relative position + World Model
- *approximate relative position* derived by combining absolute position + World Model
- *semantic location* derived by combining (potentially approximated) relative position + World Model, e.g. “at home”, “in the car”
- *absolute position history* derived by combining (potentially approximated) absolute position + time
- *relative position history* derived by combining (potentially approximated) relative position + time; e.g. includes series of scanned visual tags or proximity events caused by “Bluetooth tags”
- *geographical speed* (derived by combining a specific past time interval + absolute position history.

**World Model**

In order to generate some of the medium-level context attributes, the Context Reasoner needs to know about objects situated in the real world. For this purpose, the Context Reasoner needs to maintain a simple database. Information about objects of interest to the service applications and the Adaptive Dialogue Manager, such as the semantic label of a specific object (e.g. “car”), an object’s current absolute location (e.g. long. x, lat. y), an object’s internal state (e.g. “engine off”), have to be defined and kept up-to-date by the owner component of the specific object, i.e. a specific service application or Adaptive Dialogue Manager component.

For reasons of computation efficiency as well as for user privacy, the objects in the World Model database should be governed by at least a simple authorization schema. A world model object marked as “public” by its owner will be accessible by all other service applications, while “private” would mean that only the specific service application, which defined and inserted the object into the world model, is allowed to access the object. In this way, the service applications and the various components making up the Adaptive Dialogue Manager can share real-world contextual knowledge and reduce redundancy, while at the same time maintaining security among service applications.

In addition to the information about real-world objects, the World Model also contains information about all sensor instances known to the system.

### High-Level Context Models

There is a huge body of evidence within the area of Context Awareness, well aligned with previous related findings in Artificial Intelligence research, showing that high-level interpretations of contextual data (for example, a higher temperature in a room inevitably implies that there are more people in the room; if the person speed is higher than 10 km/h, that person is running) is very risky unless the application domain is extremely well specified. For a general-purpose architecture like the one intended to be developed in CHAT, only the application services themselves (which by definition are targeted to specific application areas) can draw high-level conclusions from context data. It is therefore left to each individual service application to construct its own, more or less complex, high-level context model if the execution of the application is improved by such a model. For this purpose, the Context Reasoner offers the service applications access to three context-related resources: the LLCM, MLCM and the World Model.

## 4 A Service Application Example: The “Gaus’ Day” Game

The proposed context modeling architecture is currently deployed in various mobile application scenarios within the CHAT project. Here we briefly describe the use of this architecture in *one* such application: the current version of the educational mobile (m-learning) game “Gaius’ Day” [1] designed to support students’ school visits at the archaeological park of Egnathia, Italy. The next section describes how we are currently enhancing the game using the proposed context modeling architecture.

The “Gaius’ Day” game is structured like a treasure hunt to be played by groups of 3-5 students: it combines the excitement of chase and solving the case with the joy of freely exploring a place and discovering its hidden secrets. The students’ challenge is to discover meaningful places in the park following some indications, and mark the position of the place on a map. Each group is given a cell phone and the map of the park on paper. The cell phone is used as an instrument to communicate the challenge and to store data about how the game progresses. Due to display size limitations, the challenge is divided into separate units, corresponding to missions.

The group walks around the ruins trying to identify the place the mission refers to (Fig. 3 left). When students believe they have identified the target place of the mission, the leader digits on the phone the numerical code of the place, indicated in the park by small signs, on the phone. A sound signals that the current mission is concluded and the next mission is beginning. It visualizes the text of the new mission and reads it out aloud to attract the attention of all group members.


**Fig. 3.** A group performing the current version of the game (left), the 3D reconstruction of the public baths visualized on the phone (center), and the existing remains (right)

After completing the last mission, the group can explore on the phone the 3D reconstructions of the places correctly identified, so that they can visually compare the possible ancient look with the existing remains (Fig. 3 center and right).

## 5 Deploying the Context Model

In the first version of the Gaius' Day mobile game architecture, all data was stored and handled locally on the cellular phone during the game phase, and the mobile system did not consider contextual aspects. On the contrary, the current version lets the cellular phone be in constant connection with a game server and takes several aspects of context into consideration to enhance the user experience both during the game phase and during debriefing. In this section we account for how the context model proposed in Section 2 and illustrated in Fig. 2 is being deployed. Apart from providing each group with one cellular phone (featuring technolgies such as GPS, Bluetooth, WLAN), we also let all students carry small limited-range Bluetooth transceivers to be able to get an approximate measure of the relative position between each student and the mobile device carried by one of the students in the same group.

In particular, the *high-level service application* "Gaius' Day" inserts the following entities into the *World Model* (see Fig. 2), described in pseudo code:

```
student:
   INTEGER studentID
   STRING student_name
   INTEGER BluetoothDeviceID

student_group:
   INTEGER groupID
   INTEGER LIST group_members (studentID)
   INTEGER game_play_status (# of missions completed)
   INTEGER mobile_deviceID

mobile_device:
   INTEGER student_owner_group
   GPS_LOCATION absolute_geographical_position

location_of_interest:
   STRING location_name
   GPS_LOCATION absolute_geographical_position

sound_source:
   STRING sound_name
   SOUND_DATATYPE sound
   GPS_LOCATION absolute_geographical_position
```

Fig. 4 illustrates the world model and some contextual properties for the Gaius' Day service application.

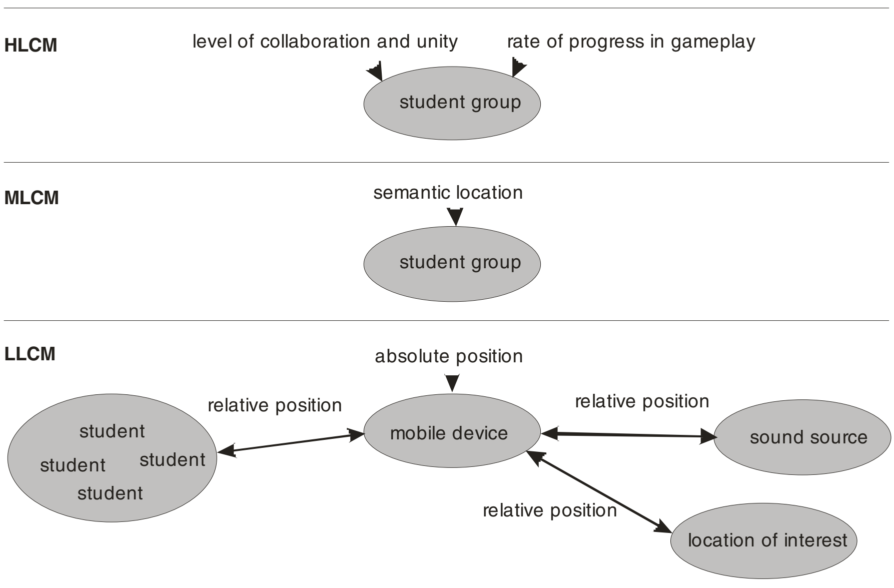



**Fig. 4.** World model entities used by the Context Reasoner module to provide low-level and medium-level context properties to the Gaius' Day service application

Specifically, the Gaius' Day service application makes use of the following LLCM properties provided by the Context Reasoner:

- *Time.*
- *Absolute position* of group's mobile device, provided by inbuilt GPS, modeled as *real-world context sensor* in Fig. 2.
- *Relative position* of students with respect to the group's mobile device, approximated through the presence or absence of a Bluetooth connection between the mobile device and the transceivers carried by all students and modeled as *real-world context sensors*.
- *Relative position* of the group's mobile device with respect to the locations of interest stored in the *World Model*.
- *Relative position* of the group's mobile device with respect to the locations of the sound sources placed at strategical positions at the site.

From the MLCM, the Gaius' Day service application makes use of the following deducted contextual properties:

- *Approximate absolute position* of each student derived by combining the relative position of each student in relation to the group's mobile device.
- *Approximate relative position* of the group's mobile device with respect to locations of interests derived by combining absolute position of the device and absolute positions of locations of interest as stored in the *World Model.*
- *Semantic location* derived by combining relative position mobile device<->location of interest.
- *Absolute position history* derived by combining absolute position of mobile device + time.
- *Relative position history* derived by combining relative position of mobile device with respect to locations of interest + time.
- *Geographical speed* of a student group derived by combining the time interval spent on a specific game mission and the absolute position history of the mobile device during that time interval.

Based on these properties given by the LLCM and MLCM, the Gaius' Day service application maintains its own high-level context model, deducing properties such as:

- *The rate of progress in gameplay* for a specific group, based on how much the group's path at the site for each mission deviates from the ideal path, how much time it takes for the group to complete missions, and how much help from the "Oracle" is requested by the group. The rate of progress could be used for adapting the Oracle help to become more elaborate for slower groups and brief for faster groups, so that the challenge faced by each group is on the right level for optimal learning efficiency and motivation.
- *The level of collaboration and unity* within a specific group, based on movement patterns of group members in relation to each other.

In addition, the three-tiered context model architecture allows for additional functional and experiental value to the Gaius' Day gameplay as follows:

- It replaces the need for entering a location code on the mobile device as the final answer to each mission, with a simple press of a button, letting the GPS coordinates gathered in the background provide the answer.

- It makes it possible for the service application to wait with audio and visual game information until the whole group is gathered (determined by relative position between students and the device).
- It enables service applications to easily create a geo-spatial sound environment by placing virtual sound sources at various locations, perceivable through headphones, and whose volume depends on the distance between GPS device and sound source. In the case of the Gaius' Day game, we are placing noise from people at the market location, crackling of fire at the furnace location, etc.), determined by the GPS coordinates of the groups' cell phones. The intention is to increase the number of cues available for problem solving as well as to enhance the overall user experience.

## 6 Conclusions

We have presented a three-tiered context modeling approach to mobile services intended to simplify design, reduce computation needs on the client side (until mobile hardware catches up), and to reduce redundancy in general. The real value and power of the proposed context modeling approach for mobile human-computer interaction can only be assessed by successfully deploying it in a wide set of real-world scenarios. We are currently performing a field study with children of a middle school in Bari playing the Gaius' Day game based on the presented context model architecture. It will allow us to better assess both the technical aspects (which has been the focus of this paper) as well as usability and user experience dimensions.

We plan to deploy the model also in other mobile scenarios to determine its generality and utility. Although this empirical validation is not yet concluded, some benefits of the approach, as presented in this paper, have been already proven in formative evaluations carried out in our laboratory.

## References

1. Costabile, M.F., De Angeli, A., Lanzilotti, R., Ardito, C., Buono, P., Pederson, T.: Explore! Possibilities and Challenges of Mobile Learning. In: Proceedings of CHI 2008 Conference on Human Factors in Computing Systems, pp. 145–154. ACM Press, New York (2008)
2. Dey, A.K., Salber, D., Abowd, G.D.: A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. Human-Computer Interaction 16(2-4), 97–166 (2001)
3. Hong, D., Schmidtke, H.R., Woo, W.: Linking Context Modelling and Contextual Reasoning. In: Kofod-Petersen, A., Cassens, J., Leake, D.B., Schulz, S. (eds.) 4th International Workshop on Modeling and Reasoning in Context (MRC), pp. 37–48. Roskilde University (2007)
4. Schmidt, A.: Ubiquitous Computing – Computing in Context. Ph.D Thesis, Lancaster University (2002)
5. Weiser, M.: The Computer for the Twenty-First Century. Scientific American, pp. 94-10 (September 1991)

# Barcode Scanning from Mobile-Phone Camera Photos Delivered Via MMS: Case Study*

Adam Wojciechowski and Konrad Siek

Poznan University of Technology, Institute of Computing Science
ul. Piotrowo 2, 60-965 Poznan, Poland
Adam.Wojciechowski@put.poznan.pl, Konrad.Siek@gmail.com

**Abstract.** Price comparison services are commonly available via WWW interface and they belong to the most popular web-sites because their functionality is highly valued by users. However, in dynamic action of shopping, customers equipped in mobile devices have no time and conditions for precise describing products on their mini screens of PDAs or mobile phones. An alternative request for information about prices and availability of a product encountered during shopping might be sent to price comparator through MMS channel in the form of photograph with product's barcode. In the paper we describe a series of experiments using mobile phones for sending camera-made barcodes and deriving numerical equivalence for product ID. Experiments were conducted on a set of barcodes in ideal and real conditions using a library for software barcode scanning. Collected results show two main sources of difficulties in using such a system: human-related problems and technical issues described in the paper. A separate issue is availability of barcode id in product description stored in price comparator database.

**Keywords:** Mobile-phone camera photos, barcode scanning from photo, MMS.

## 1 Introduction

Price comparison sites are popular among Internet users because they provide valuable functionality in a very simple form. It is a common practice for products to be identified by name and associated descriptive attributes. In the simple form, searching algorithms are based on selection of database records where user defined phrase is a substring of product name or product's description. The tool is oriented towards supporting intentional shopping, where users know the product.

The value of information provided by a price comparator has two aspects. Firstly, it is a price ranking which allows the customer to select the cheapest offer for particular product. The other benefit for the customer is increased confidence. The attribute is closely related to number of offers presented to the customer. Even if the customer does not decide to choose the cheapest offer, he may know how far from the optimum the price of the preferred offer is. High number of product providers and price offers

* This research was partially supported by the grant 91-439/BW.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 218–227, 2008.

for particular product is considered a measure of market penetration. The more offers are displayed on user's screen the better confidence he gets that his selection process is based on rich knowledge of market reality. This functionality, *increased confidence*, may be especially valued by mobile customers, who encounter a product in the market, put it in their shopping cart, but wonder if the price they see on product label is acceptable in the sense that it is *close enough* to optimal offers available for the customer in other shops. The understanding of *close enough* to optimal price is a function of particular shopper's preference model and circumstances in which he is shopping. A typical window-shopper is more likely to make analytical calculations and select another product provider (e.g. an e-shop) who can offer the same good at a better price than a person quickly looking for a swimsuit for tomorrow's training 10 minutes before the shop is closed. In the swimmer's case the price and it's distance from the optimal price offer for this product is less important than the need to obtain the product quickly.

A convenient access to price comparators is available via a WWW interface, usually designed to be seen on big computer screen. Even though mobile users equipped in PDAs or mobile phones have the possibility to read web-pages on their devices the interaction through small keyboard is difficult and error-prone. Quite often, in dynamic action of shopping, mobile customers have neither the time nor conditions for precise description products on their mini screens of PDAs or mobile phones. However, the devices are commonly equipped with a photo camera. An alternative request to price comparator application for information about prices and availability in various locations of a product encountered during shopping might be sent to a price comparator through MMS channel in the form of a photograph with the product's barcode. System's response sent to mobile device would provide the top of price ranking from offers for a product defined by its barcode. An approach, based on a social network as an information feed, to keep the data stored in price comparator up-to-date was proposed in [1].

Another use of barcode scanning from photographs delivered via MMS my be quick product request or referral marketing. An interesting business model for a new form of word-of-mouth referral marketing based on RFID (Radio Frequency IDentification) tags scanning was proposed in [2]. The current reality of European shops is that almost every product is labeled by a 1D barcode, while RFID tags are still in experimental use. Word-of-mouth marketing, where recommendation to buy particular product comes from a trusted person, can be supported by scanning a product's code and sending a request for an identical item to suppliers shopping system. Ordinary mobile phones are not able to scan barcodes but they are equipped with a digital camera. Making a photograph of a barcode label on sought product and sending it via MMS is much easier than searching for the identical product in text based price comparators. A similar scenario of buying products by sending a photograph of the product's barcode may be applied when the barcode is displayed on a billboard and people passing-by can make a picture of the code and order the product or express interest in particular product which is sent to seller offering the product. The request (barcode picture) sent via MMS identifies the product and phone number of the person who sent the request – an excellent information for merchandiser willing to sell the product and a very convenient way of shopping for the customer. This marketing channel becomes more and more explored, because *the cellphone is the natural tool to combine the physical world with the digital world* (C.Roeding cited in [3]). Barcode stored on mobile phone in form of an image (either photographed or sent to device in MMS) and displayed on phone's screen can be scanned. It opens a gate to use the technology to sell electronic tickets which could be validated by existing barcode scanners.

In the paper we collect our experience from experiments we performed on the way to develop an application capable to read product id from a barcode photo sent via MMS. Software developed in our experiment is available at [4] and everyone can run his own similar experiments. The software can also be used in training the user in making a photograph of a barcode in order to make the image readable for the scanner. Problems encountered in our experiments may be divided into two sets: human-related and technical issues. In the context of applying the system to a price comparison system a separate issue is availability of barcode id in product description stored in price comparator database.

## 2 Tools Integrated in Barcode Scanner Used in Experiment

*JBiedronka* [4] is a Java application developed at Poznan University of Technology, Poland as a desktop program which can be used to simulate software scanning of barcodes provided on images stored in JPG or PNG files. The software was developed as a student project with intention for experimental use. The program integrates functionality provided in *J4L RBarcode Vision* [5] library and *ImageMagick* [6] in order to scan the barcode and enhance quality (if necessary) of the image.

The use of the *J4L RBarcode Vision* library for extracting barcodes textual equivalence from images was decided on several factors:

- the library provided support for a wide range of one-dimensional barcode type, which are most commonly used throughout the world for coding information about commercial products. i. e., the EAN-13 and EAN-8 are the recognized standards for GTIN (Global Trade Item Number) [7].
- at the time of development (Spring 2007) research over the Internet resources gave no reliable open-source libraries, so it was decided to use a trial version of a commercial product for the study. The chosen option has the disadvantage of providing no means of applying changes derived from the experiments, however, it does provide an advantage of being a library, which is used commercially in real-life situations, and, therefore, an apt target of scrutiny.
- additionally, a version of the library has been prepared to function with Java, which well suited the non-functional requirements for the project, as barcode scanning was supposed to be performed in that language.

In addition to the barcode scanning software a graphical library has been used to provide digital re-mastering, if it were to cause the images to be scanned correctly. The *ImageMagick* suite was selected for this purpose, as a portable, open-source solution with a command-line interface.

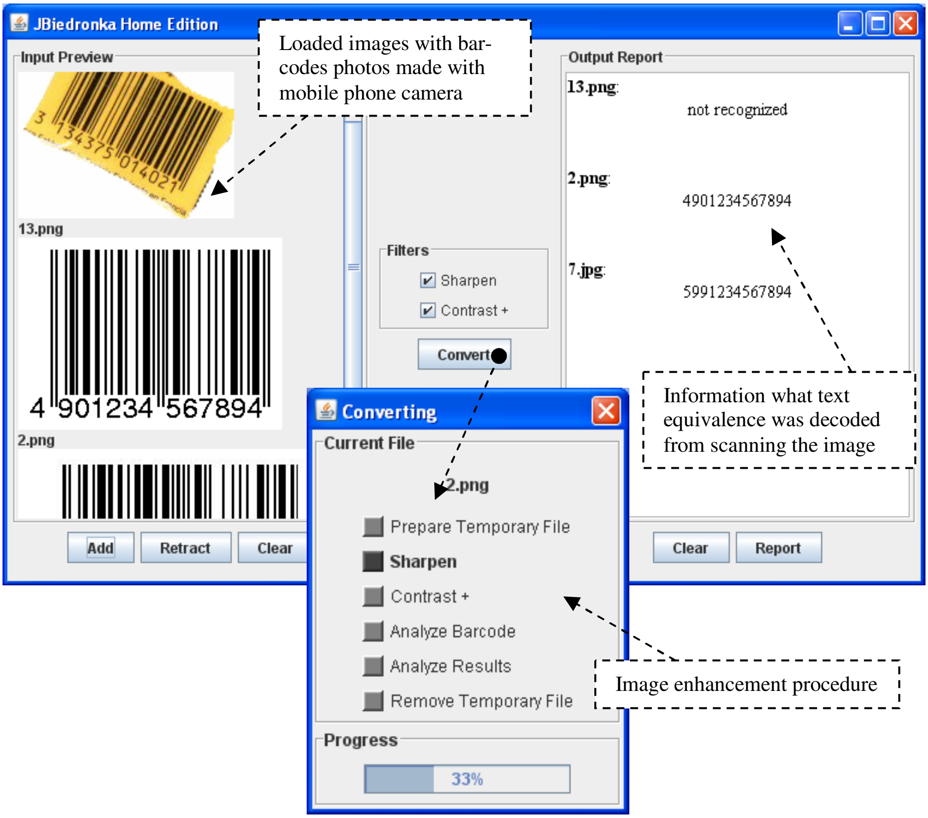



**Fig. 1.** *JBiedronka*: barcode image scanner designed for the experiment

## 3 Examined Objects

### 3.1 Camera Items

A number of images gathered from various sources and of different characteristics were processed for recognition during the study. Similar cases have been grouped into sets, having similar characteristics and results. All the files mentions below are collected in [4] for evidence and comparison.

The most significant group of images that had undergone processing were the images gathered from mobile phone users, who were asked to take pictures of barcodes while in shopping venues. The users were not trained to how to make a good quality picture. Our intention was to observe how people understand *catching* a barcode on a photograph. In response we obtained a set of images, among which some were the photos of a barcode itself, there were photos of entire product with a barcode label and the most substantial part were photos where the barcode was in central part of the image but it was surrounded by graphics and text printed on given product (see fig. 2).


**Fig. 2.** Sample real photographs of barcodes made with cameras built-into mobile phones

The images were done using three models of mobile devices: Sony-Ericsson K510i (images real/sek510/*.jpg,) Motorola K1 (images real/mk1/*.jpg) and Motorola C651 (images real/mc651/*.jpg). This presented the most realistic cases might be expected were the service implemented for actual commercial use. Nonetheless the recognition failed in every presented instance.

Aside of not optimistic results of barcode scanning from photographs taken by mobile-phone camera, we faced a new problem: the picture sent by mobile user may contain much more than just a barcode. Our analysis let us to conclusion that people tried to have the barcode sharp on the picture, they naturally felt that sharpness of lines is a key issue in recognition process. However, mobile-phone cameras used in experiment offered no *macro* function [8] and it was not possible to make a sharp picture of small and close objects. In order to have the image sharper the users took pictures of objects at some distance which resulted in catching the barcode and its surrounding. Other problems identified in this experiment, that could be attributed to human skills (missing training) include: light reflects on barcode, barcode was not flat and taken under some angle which effected with perspective on the picture.

Another attempt was made with another group of images taken with a mobile phone camera of a Sony-Ericsson K510i (images filter/sek510/*.jpg). These instances were presented an improved quality to the previous group by applying a black-and-white filter to the images at capture and supplying improved lighting conditions, the images, however, were still heavily blurred. The results of recognition of this group were unsatisfactory – no images were recognized correctly.

### 3.2 Synthetic Items

Following the results presented in chapter 3.1, cases were prepared to study the individual characteristics of pictures which influence the process of recognition.

Twenty images (see fig. 3) were presented to the barcode recognition software, which were obtained from a scanner device or computer-generated (images img/2.png, img/3.png, img/6.png, img/7.png and img/12.png) and present high quality. The barcodes are uniformly black, placed on white background, lean to no angle and present easily distinguishable, undisrupted straight lines. Such barcodes present an ideal case for the recognition software and produce good results.

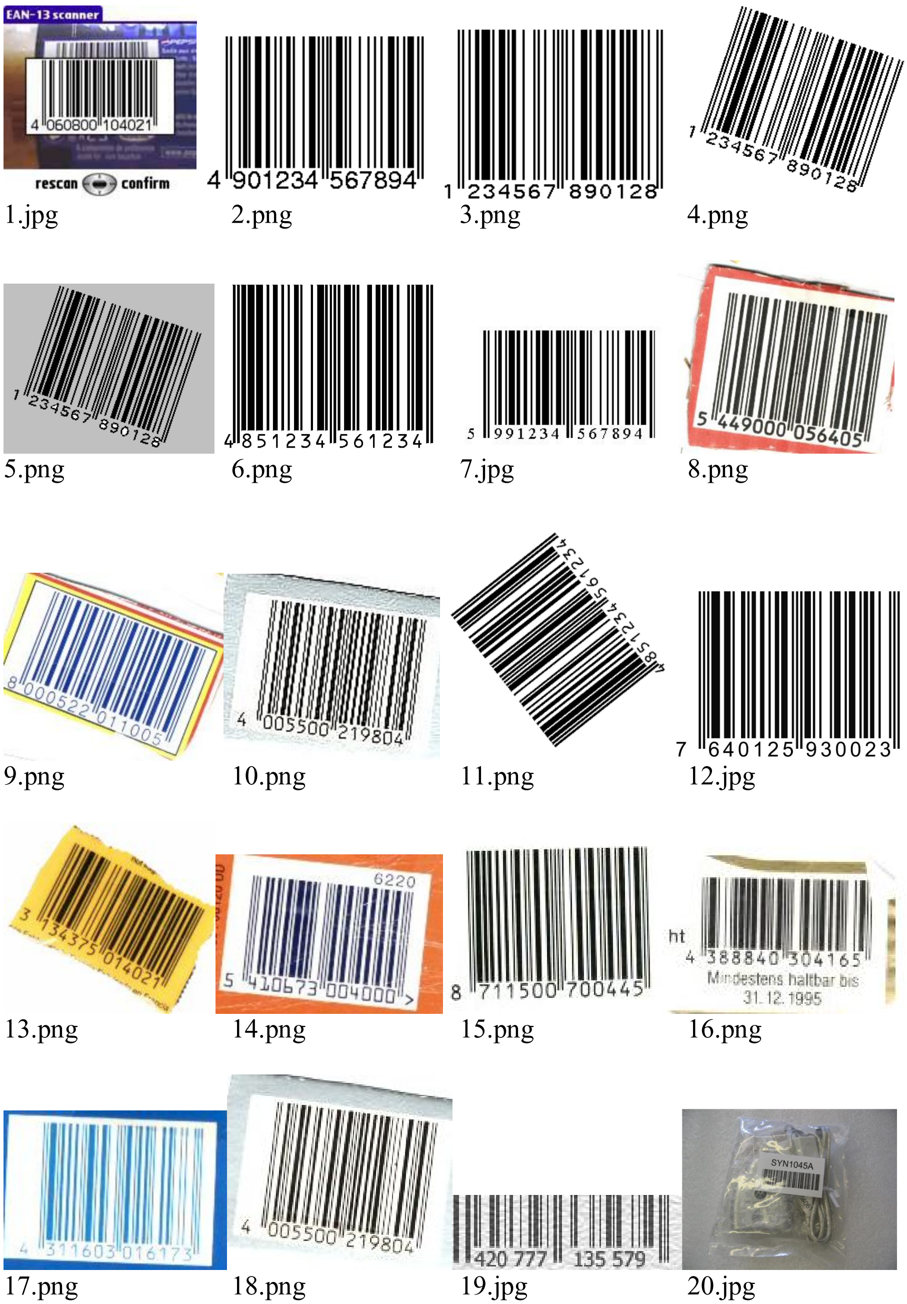



**Fig. 3.** Barcode pictures used in study of factors affecting quality of recognition

An additional number of images obtained from a scanner device were presented for recognition, at this instance, with a skew ranging from about 45 degrees (img/4.png, img/10.png and img/13.png) to over 90 degrees (img/11.png.) The images img/4.png and img/10.png were recognized correctly, while img/13.png and img/11.png remained unrecognized. The evident cause of the lack of recognition of img/13.png seems to be the slight blur of the bars, visible especially well in the center section of the image resulting from the skew. The failure in reading img/11.png is apparently caused by rotation, which rendered the reader unable to locate the barcode.

A different set of images of similar origin to the one presented above were tested, however, these images include a multi-colored background, which results from being placed on an actual item (images img/1.jpg, img/5.png, img/8.png and img/9.png.) A skew may or may not be present at these instances. The barcodes from this set were all recognized correctly.

Next, a set of images were presented for recognition, which included a slight blur to the bars (images img/15.png, img/17.png, img/18.png) or different colors of bars (images img/14.png and img/16.png) in conjunction with some of the previous features. The results were unsatisfactory within this group as all the barcodes remained unrecognized.

Finally, an image with introduced noise was processed by the recognition software (image img/19.png.) The noise applied to the image took form of groups of pixels in the color of the bars dithered across the image. The recognition of such an instance proved to be impossible by software used in our experiment.

### 3.3 Digital Re-mastering

Following the processing mentioned above two graphical enhancements were applied to the processed files in an attempt to improve the results of recognition. The remastering took form of applying the 'sharpen' function from the *ImageMagick* library followed by the conversion by raising the contrast of the image (see fig. 1). The former was meant to decrease the blurriness of the images, while the latter was to additionally resolve any problems with light. No significant improvement was noted, except for img/14.png, which, after application of both processes, became recognizable, despite the fact, that initially the blur caused otherwise.

Additionally, sharpening and contrast enhancement should be conducted with care, as some images may become unrecognizable only afterwards (i.e. img/8.png.)

## 4 Summary of the Experiment

The data used for the study can be divided into two main categories: high quality and low quality images:

- The high quality images would be those obtained primarily by the use of a scanner or barcode generation software. Although it is possible for them to be skewed at large angles, these images are typically sharp, well focused and devoid of problems caused by bad lighting conditions, which are present in the other examined group. Consequently, the high quality images have had the best chances of being read correctly.

- The low quality images would be those obtained via mobile phone cameras; the images obtained this way present several additional characteristics: their overall quality is dependant on lighting conditions, types of photographed objects and the distance of the camera from those objects. Furthermore, they have varying levels of sharpness, contrast and focus. The barcode scanning software has additional obstacles to overcome during the scanning process, and, therefore, is less likely to succeed.

The reason why barcode photographs obtained from mobile-phone digital cameras belong to *low quality* set and they are hard to derive alphanumerical equivalence of barcode is missing *macro* function in cameras used in our study. Lack of macro function in cameras built-in mobile devices is a common problem. However, the *macro* feature may become available pretty soon, when improved cameras are used in mobile phones. It slowly becomes reality (e.g. Nokia N73 has capability of making good photographs of objects at a distance of 10cm).

## 5 Interpretation of Results

Having inspected the results of the processing it is evident that its outcomes are not satisfactory to the needs of application of the system. Moreover, it can be seen that several factors are responsible for causing the inefficiencies. These factors stem from the technical aspects of the system, the faults of human operators of the cameras and the properties of the photographed objects.

Examining the ideal cases with incrementally added faults seems to indicate that the barcode recognition software can in fact overcome many interfering detail and still be able to present the correct codes. Some particular types of interference, however, pose serious problems – specifically, any blur, noise or a large rotation. While the latter problem can be easily overcome by instructing the operators of the cameras to try to align the barcode correctly, the previous two originate from the limitations of the cameras themselves. Particularly, mobile phone cameras don't seem to be able to take pictures of objects close-up, but are instead preset to taking pictures at a certain distance, which, in turn, is too far for the barcode to be legible. Additionally, the cameras tend to generate an amount of noise in case of bad lighting conditions, which prevents the bars of the barcodes to retain their widths in the eyes of the recognition software.

Moreover, the operators of the mobile phone cameras might cause problems in barcode recognition by inexpertly taking pictures of barcodes from too great a distance, at an angle to the surface of the object (which causes the relations among the widths of bars to be disturbed) or surrounded by too many details.

Furthermore, there exist problems caused by the environment, such as the lightning, or photographed objects themselves, such as the shape of the objects (for instance, round and spherical objects disturb the proportions of bar widths on a picture) or its texture (a glossy surface is bound to reflect light, so causing reflections in form of white stripes to appear on the barcode). These issues, however, are less visible in this study, as the process did not yet reach that phase, where this would become relevant, and, additionally, are a visible problem in other barcode scanning systems, not only those, which try to apply the mobile phone cameras to the process.

Finally, it is worth to mention, that digital re-mastering can be used to rid images of small blurs or lack of contrast, and may be tried if an image is not recognized, however, it should not be expected to fix any major flaws in the image. The extent to which sharpening and increasing contrast works is hard to determine, especially automatically by the recognition program, however they should not be performed on all images, as this might cause corruption on images, which would otherwise have been recognized correctly.

## 6 Application of Barcode Via MMS in Marketing

Our experiments showed that commonly used mobile phones equipped with a camera without *macro* function are not able to produce images of barcodes in quality good enough to automatically decode the image and send response to the mobile user. However, using the same camera built-in mobile phone, it is possible to make a sharp photo of a larger object. It could be a barcode but presented in such a size that a distant photo can catch the image in condition ease to decode alphanumerical equivalence of the image. Our additional experiments showed that barcode image printed on A5 paper size are big enough be taken with mobile-phone camera and correctly interpreted by our software scanner. It opens a gate to provide customers with possibility of taking a photo of a barcode presented on a billboard or printed as an ad in a newspaper. It is easy to make a picture and send it via MMS to express customer's interest in particular product or use this communication channel for registration.

Customer's query sent from mobile device in form of an MMS containing a barcode image identifies owner of the device (mobile number) and product. It is possible to automatically reply such customer by sending him an offer corresponding to request. The system receiving customers' queries may also sort and schedule requests and pass them to personal customer assistant who may contact the customer.

When cameras built-in mobile phones become capable to make sharp photographs of as small objects as barcodes printed on products, the service of mobile price comparison should become available via MMS interaction. It is an easy and convenient way for sending queries, however, what came from our experiments, users must be aware of factors affecting possibility of proper image decoding. They are, for instance: light reflects and perspective on the image. Users require training and our software: *JBiedronka* available at [4] may be used for testing whether images of barcodes coming from a camera are readable for software scanner.

## 7 Conclusion and Suggestions for Improvement

The results seem to indicate that the application of mobile phones in the process of scanning barcodes is riddled with various, non-trivial difficulties, which stem from many areas. There are technical problems, human-related problems and problems related to the photographed object or its immediate environment, each of which need to be battled in their own unique ways.

There are several ways to overcome the technical problems related to mobile phone barcode scanning, and these draw either on the improvement of the camera used in mobile phones by way of allowing them to focus on objects close-up and improving their reaction to low light, or on the improvement of the technology of barcodes. The latter is applied sporadically throughout the world for coupons, where large two-dimensional barcodes are used instead of small one-dimensional ones, which provide more and better quality information for the recognition software. However, such an approach may prove difficult to implement for a system, which should be used for gathering information about shopping, which is already identified by one-dimensional codes.

The solution to human-related problems should require simple instruction, on how to supply better quality images of barcodes via a mobile phone camera. Whether such instruction were possible to conduct or effective it is difficult to determine, however it would, with high probability, not need to be excessively long or detailed, resulting in its increased efficiency, and just a few simple details are bound to enhance the quality of the image greatly (as the Pareto rule lets one believe.) Possibly, such instruction would also be useful in overcoming the problems related to the environment as well.

## References


1. Wojciechowski, A.: Supporting Social Networks by Event-Driven Mobile Notification Services. In: Meersman, R. (ed.) OTM-WS 2007, Part I. LNCS, vol. 4805, pp. 398–406. Springer, Heidelberg (2007)
2. Lee, K.J., Lee, J.C.: Design of Ubiquitous Referral Marketing: A Business Model and Method. In: Bauknecht, K., et al. (eds.) EC-Web 2006. LNCS, vol. 4082, pp. 102–111. Springer, Heidelberg (2006)
3. Story, L.: New Bar Codes Can Talk With Your Cellphone, New York Times (April 1, 2007), http://www.nytimes.com/2007/04/01/business/01code.html?pagewanted=2&_r=3&ref=business
4. JBiedronka Software (May 2008), http://www.cs.put.poznan.pl/awojciechowski/papers/2008/M2AS/JBiedronka.zip
5. J4L RBarcode Vision (May 2008), http://www.java4less.com/vision/ vision.php
6. ImageMagick (May 2008), http://www.imagemagick.org
7. Global Trade Identification Number Information (May 2008), http://www.gtin.info
8. Macro photograpy, Wikipedia (May 2008), http://en.wikipedia.org/wiki/Macro_photography

# A Qualitative Study of the Applicability of Technology Acceptance Models to Senior Mobile Phone Users

Judy van Biljon[1] and Karen Renaud[2]

[1] School of Computing, University of South Africa, Pretoria, South Africa
vbiljja@unisa.ac.za
[2] Department of Computing Science, University of Glasgow
Glasgow, Scotland
karen@dcs.gla.ac.uk

**Abstract.** This paper investigates the factors that influence mobile phone adoption by the older user. Technology adoption is a process, with the adopter progressing from a state of ignorance of the technology to embracing it and considering it a necessity. Full progression can only occur if the adopter fully *accepts* the technology. If not, he or she is unlikely to progress towards whole-hearted adoption and remain a reluctant user or discard the technology altogether. Many theoretical models explain the dynamics of technology acceptance by proposing particular predictive factors and are based on quantitative studies built on the responses of students or economically active adults. This begs the question: *Do existing technology acceptance models incorporate the factors that lead to mobile phone adoption and use by older adults*? We consulted findings from studies of senior mobile phone users and extracted a number of issues concerning needs, uses and limitations, which we verified by means of structured interviews with senior mobile phone users. We compare these qualitatively derived issues with the factors from existing quantitative models. This led to the identification of a two-dimensional adoption matrix where verified acceptance factors, derived from the experiences and opinions of our participants, are mapped against a recognised adoption process, highlighting the fact that current models only partly predict adoption and acceptance by the senior mobile phone user.

**Keywords:** Mobile, technology acceptance, usability, elderly.

## 1 Introduction

This paper addresses technology acceptance and use in the context of the senior mobile phone user. These users are an oft neglected group in product development and marketing, yet in most developed societies they are the only growing age group [1]. Technology acceptance, in general, has been widely studied and several models of technology acceptance have been proposed and tested [2, 3]. In the field of Management Information Systems (MIS), a number of *technology acceptance models* have been proposed, which focus on *factors* influencing acceptance (without considering the process) [4]. Sociological studies go beyond the purchasing decision to describe technology's acceptance, rejection and use i.e. the adoption *process* [5]. Lee [3]

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 228–237, 2008.

suggests a pragmatic approach that allows a merging of different approaches, and this is the approach we will be using in this paper since it enhances understanding of *factors* influencing progression through the different *adoption* phases.

The current acceptance models have been quantitatively verified, using students or economically active adults as participants [3]. This begs the question: *Do the factors from the models based on responses of students or people in their working environment predict mobile phone acceptance by the elderly*? (We use the terms senior, elderly and older adult to refer to people over the age of 60). A literature study of the needs, limitations and expectations of the older adult mobile phone user made it clear that they demand added value in the form of a more social, active, meaningful and independent life [1]. In terms of simple economics, the value of the phone can be expressed as *value = f(usefulness, ease of use).* How each factor is rated depends on each person's individual needs (perceived usefulness) and abilities (perceived ease of use) and it is difficult to come up with a definitive way of measuring these. However, knowing which factors *mediate* ease of use and usefulness could provide insight into the concept of value. The question being addressed here is whether current mobile phones deliver this value to elderly users.

We address this question by attempting to verify, *qualitatively*, the factors included in the quantitatively derived models, and their influence in the adoption process. Firstly, we need to identify a set of factors relevant to mobile phone acceptance and use. We constructed five scenarios related to senior mobile phone usage and presented these scenarios to participants in semi-structured interviews. The participants' responses were used to validate or deny the identified factors. Our findings are presented in an *acceptance matrix* which maps the validated *factors* against the domestication technology *adoption progression*. This helps us to understand the factors that play a part in leading users down the road towards whole-hearted adoption of a particular technology.

Section 2 lays the theoretical groundwork by presenting an overview of technology acceptance research and extracts common factors from the various models. Section 3 describes the context of the elderly mobile phone user. Section 4 outlines the qualitative study and presents the results. Section 5 contextualises the findings and concludes.

## 2 Technology Acceptance Models

Technology acceptance has been studied by researchers in different fields. On the one hand researchers have proposed various levels of progression from not having used the technology to full adoption. Rogers' innovation diffusion model [8] and the domestication approach are examples of these. Rogers' innovation diffusion model focuses on marketing and sales processes, the domestication approach deals with a more global analysis of acceptances *ex post facto.* Another perspective, from the field of Management Information Systems (MIS), identifies *factors* that influence technology acceptance. Examples are the Technology Acceptance Model (TAM) [6] and the Unified Technology Acceptance Model (UTAUT) [2] which focus on technology acceptance within organisations [5,7]. Sections 2.1 and 2.2 reviews these models and then Section 2.3 discusses the application of technology acceptance models to mobile technology and extracts a set of factors relevant to mobile phone acceptance by senior users.

### 2.1 Technology Diffusion - Processes

Two very different models, which depict the technology adoption process, are presented by Rogers [8] and Silverstone & Haddon [9]. Rogers[8] proposed a five stage process of product adoption: the knowledge phase where the person gets to know about the product; the persuasion phase where he or she becomes persuaded of a need for the product; the decision phase which leads to a purchase; the implementation phase where the item is used and the confirmation phase where the individual seeks to confirm that he or she made the right decision in purchasing the product.

Silverstone and Haddon [9] proposed the *domestication of technology* as a concept used to describe and analyze the processes of acceptance, rejection and use as described in Table 1. Users are seen as social entities and the model aims to provide a framework for understanding how technology innovations change, and are changed, by their social contexts. The domestication theory adoption process is more suitable for our purposes in charting acceptance factors since it is less marketing oriented than Rogers' model.

**Table 1.** Four dimensions of domestication adoption process [3] :66

| **Dimension** | **Description** | **Examples of potential themes relevant in user experience research** |
|---|---|---|
| Appropriation | Process of possession or ownership of the artifact. | Motivation to buy a product<br>Route to acquire information about a product<br>Experience when purchasing a product |
| Objectification | Process of determining roles product will play | Meaning of a technology<br>What function will be used in users' life?<br>Where is it placed? How is it carried? |
| Incorporation | Process of interacting with a product | Difficulties in using a product (usability problems)<br>Learning process (use of instructional manual) |
| Conversion | Process of converting technology to intended feature use or interaction | Unintended use of product features<br>Unintended way of user interaction<br>Wish lists for future products |

### 2.2 Technology Acceptance - Factors

The Technology Acceptance Model (TAM) propose a number of factors that are essential in determining user attitude towards accepting a new technology, as shown in Figure 1. [6, 10]. TAM incorporates six distinct factors [6, 11]:

- *External variables* (EV), such as demographic variables, influence perceived usefulness (PU) and perceived ease of use (PEU).
- *Perceived usefulness* (PU) is defined as 'the extent to which a person believes that using the system will enhance his or her job performance' [2].

- *Perceived ease of use* (PEU) is ‘the extent to which a person believes that using the system will be free of effort’ [2] .
- *Attitudes towards use* (A) is defined as ‘the user’s desirability of his or her using the system’[10]. Perceived usefulness (PU) and perceived ease of use (PEU) are the sole determinants of attitude towards the technology system.
- *Behavioural intention* (BI) is predicted by attitude towards use (A) combined with perceived usefulness (PU)
- *Actual use* (AU) is predicted by behavioural intention (BI).

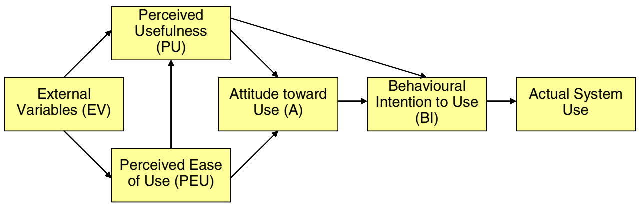



**Fig. 1.** Technology Acceptance Model (TAM) [10]

Venkatesh et al. [2] extended TAM and developed the Unified Theory of Acceptance and Use of Technology (UTAUT), which attempts to explain user intentions to use an information system and subsequent usage behaviour. An important contribution of UTAUT is to distinguish between factors *determining* use behaviour namely the constructs of performance expectancy, effort expectancy, social influence and facilitating conditions and then factors *mediating* the impact of these constructs. The mediating factors are gender, age, experience, and voluntariness (i.e. the degree to which use of the innovation is perceived as being of free will). Both TAM and UTAUT can be applied to any technology type but there is some value in specialising the models for particular technologies. The following section discusses the application of models in the mobile technology area.

### 2.3 Specific Models for Mobile Technology Acceptance

Kwon and Chidambaram [12] propose a model for mobile phone acceptance and use which includes the following components: demographic factors, socio-economic factors, ease of use, apprehensiveness, extrinsic motivation (perceived usefulness), intrinsic motivation (enjoyment, fun) social pressure and extent of use. They found that perceived ease of use significantly affected users' extrinsic and intrinsic motivation, while apprehensiveness about cellular technology had a negative effect on intrinsic motivation [12]. The limitation of this model is that it does not include infrastructural factors, which are essential in mobile technology. The Mobile Phone Technology Adoption Model (MOPTAM), depicted in Figure 2, integrates TAM with the determining and mediating factors from UTAUT and then adapts the result to model the personal mobile phone use of university students.

Based on exploratory research, Sarker and Wells [13] propose a framework that relates *exploration* and *experimentation* to the *assessment of experience* that determines *acceptance outcome*. The mediating factors distinguished are: context, technology characteristics, modality of mobility, communication/task characteristics, individual characteristics. Table 1 summarises the factors incorporated into the different models. It is interesting to note that perceived ease of use and actual use are common to all.

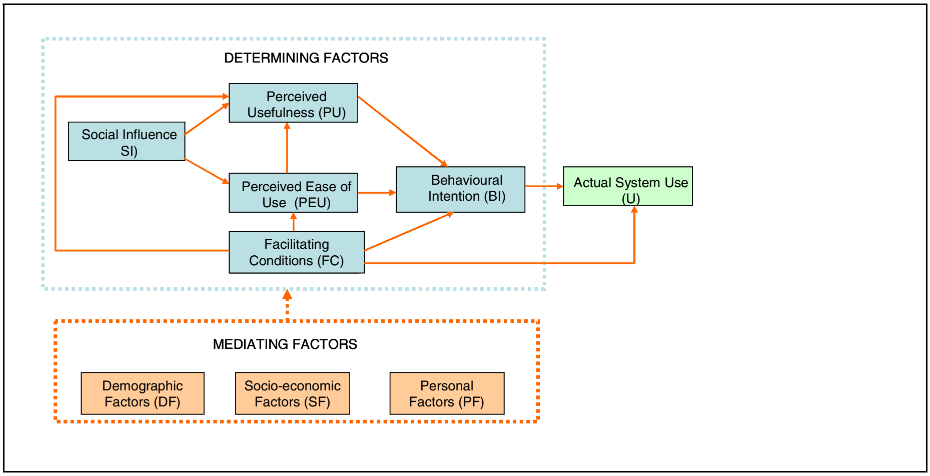



**Fig. 2.** Diagrammatic representation of MOPTAM[4]

Table 2 summarises the most fundamental factors incorporated into the models listing the following factors: Social influence (SI), Perceived Ease of Use (PEU), Perceived Usefulness (PU), Facilitating Conditions (FC), Behavioural Intention (BI), Demographic Factors (DF), Socio-Economic Factors (SE) and Personal Factors (PF). Note that perceived ease of use is the common factor across all the models.

**Table 2.** Factors influencing mobile phone acceptance

| | Models and theories | | | | |
|---|---|---|---|---|---|
| Factors | TAM | UTAUT | Kwon & Chidambaram | Sarker & Wells | MOPTAM |
| SI | No | Yes | Yes | Yes | Yes |
| PEU | Yes | Yes | Yes | Yes | Yes |
| PU | Yes | No | No | Yes | Yes |
| FC | No | Yes | No | Yes | Yes |
| BI | Yes | Yes | Yes | No | Yes |
| DF | External variables | No | Yes | Yes | Yes |
| SE | External variables | No | Yes | Yes | Yes |
| PF | No | No | No | Yes | Yes |

## 3 Context of the Senior Adult

The mobile phone needs of the elderly centre around critical services such as emergency and health support that enhance safety and those services that make everyday life and tasks easier [1, 3]. The elderly position mobile phone use in terms of *value,* which is mostly based on communication and safety aspects. In terms of the factors identified in Section 2.3, the senior mobile user is positioned as follows:

**Social influence(SI):** Phillips and Sternthal [14] mention that with increasing age comes reduced involvement with others as confirmed by Abascal and Civit [15]. The reasons are argued by experts, but the net effect is unarguable: reduced access to information that is readily available to younger people. The social contact they do have is primarily with their extended family and this group appears to provide them with the advice and support they need. Friends and relatives, especially the opinion of children and grand-children impact the behaviour of the elderly mobile phone user [1, 3].

**Perceived ease of use (PEU):** The undeniable potential of the 'gray' market is hampered by the physical and cognitive limitations of aging. Older mobile phone users make use of fewer mobile phone features than younger users [3]. Ziefle and Bay [16] suggest that older mobile phone users do not have a mental model of the ubiquitous hierarchical menu system used by mobile phones. They struggle to find the features they want to use and therefore do not use them. This is confirmed by a study carried out by Osman et al. [17] who interviewed 17 older users and asked them to name the most important features of a mobile phone. "Easy menus" was mentioned most often, followed by large screen. The latter is unsurprising since many older users have impaired vision. Another factor mentioned is that they require large buttons, due to the inevitable decrease in manual dexterity experienced by many older users [3].

Regarding perceived usefulness (PU), senior mobile users use mobile phones in the following areas:

**Safety and security:** Kurniawan [18] claims a security-related focus drives the mobile phone market for older users. She says that older people use phones to make them feel less vulnerable when alone at home and more confident going out alone. They used their mobile phones mostly in emergencies or to contact family members.

**Organisation:** The mobile phone offers the potential for organising and synchronising activities in new and cost effective ways. This organisation can be of a personal nature, such as using the reminders to compensate for memory failure, or the phone book to store telephone numbers.

**Information:** The number of seniors surfing the Internet is steadily increasing [17], evidence of their need for, and readiness to use, the Internet to get information they require. The use of a mobile phone to access the Internet is an obvious next step for all users, and the elderly are no exception.

The remaining factors in Table 2, such as facilitating conditions, are all mediating influences [1, 3]. Mallenius *et al.* [1] argue for using functional capacity (consisting of the physical, psychological and social aspects), rather than age as a facilitating condition. Furthermore, regarding infrastructure, it is well known that the elderly consider spending very carefully and the price of a device or service is a differentiator for use [1, 4].

## 4 The Qualitative Study

Thirty four elderly people participated in our study (10 male and 24 female). The participants per age distribution were: 60-70 years: 13, 70-80 years: 16 and 80-90 years: 5. Considering mobile phone use, 19 of the participants had contracts and 15 used pre-pay. They obtained the phones by buying them (16), having the phone bought by someone else (3) or getting the phone as a gift (15). The majority who bought the phones themselves were in the 60-70 age group, i.e. the younger participants.

Five scenarios seniors typically encountered in their everyday lives were presented to the participants and they were requested to comment. The participants responded actively, providing the following information:

**Scenario 1** *(obtaining information about mobile phones):* Relating to information gathering, the responses fell into three groups: 9 said that people would ask their children; two said that they should ask people of their own age(not their children); while 23 reckoned that people would go to mobile phone vendors for information.

**Scenario 2** *(accepting a cast-off phone)*: Relating to the impact of decreased ability to learn versus other motivations, three main groups of responses arose: 11 answered yes, citing the *economical* 'You can sell the old one'; the *philosophical* 'You should take the challenge'; and the *pragmatic*: 'The old phone may be getting out of date' as reasons. Seventeen answered no, stating memory loss and difficulty in learning as reasons. A third group of 6 reasoned that it depended on the person and the circumstances.

**Scenario 3** *(emergencies such as having a stroke):* Relating to safety and ease of use, 21 participants said that a mobile phone could be useful in emergencies, 12 felt that the older person would be 'too scared and confused', or 'unable to find spectacles'. The rest felt that theoretically it was a good idea, but not practical since older people find phones difficult to use, even more so when stressed.

**Scenario 4** *(accessory in helping to remember)*: Relating to the need for organisation, 28 reckoned that people could set a reminder, of these five had reservations about whether older people would manage to do that, whilst one was unsure that a mobile phone would be of any help.

**Scenario 5** *(safety aid in travelling):* Relating to safety and usefulness the majority(27) agreed that it could be useful, they gave different reasons such as the traveller contacting (phone or SMS) the family or vice versa, while some believed it could be used by a third party in the event of an emergency.

It is clear that *ease of use* cannot really be self-reported. It is far more enlightening to observe users making use of a product. We therefore asked participants:

- To name the three features they used most often, and
- To show us how their phone performed the features.

We intended to count the button presses in the second part of this process in order to gauge effort expended and consequent ease of use. We had to discontinue this since it became obvious that the participants had difficulty finding their most-used features. Some participants asked the experimenter for assistance in finding the feature, others tried various routes down the menu structure before triumphantly locating the desired feature. We felt that the button press count was so inaccurate as to be useless and therefore discontinued counting. Since the type of mobile possessed by the participants was diverse the unavoidable conclusion is that most phones have serious ease-of-use issues, as least as far as our demographic was concerned.

Table 3 relates the acceptance factors indentified during the interviews to the domestication dimensions to form and adoption matrix. Note that no evidence of *conversion phase* activities was uncovered while social influence is prevalent in all of the first three stages. For this group, perceived usefulness (PU) and perceived ease of use (PEU) are often not considered in the appropriation phase since many of the elderly do not select their own phones. This gap in appropriation as depicted in Table 3 may provide an explanation for the observed lack of acceptance. Usefulness and ease of use are considered in the objectification phase and incorporation phases where some concerns about usefulness, and serious concerns about ease of use, were identified.

**Table 3.** Adoption matrix: Relating domestication progression to factors

| | Factors | | | |
|---|---|---|---|---|
| Dimension | SI | PU | PEU | BI |
| *Appropriation* (Scenarios: 1,2) | Yes | No | No | No |
| *Objectification* (Scenarios 2,3,4) | Yes | Yes | Yes | Yes |
| *Incorporation* Identified in observation | Yes | Yes | Yes | Yes |
| *Conversion* | No evidence in observation or interviews | | | |

## 5 Discussion and Conclusion

The applicability of TAM for the elderly mobile phone user has been investigated in this paper. In the light of previous findings, we are interested in the relative impact of various external factors, usefulness and ease of use on attitude and intention to use.

Several researchers have tested the applicability of TAM to specific groups [19]. However, the elderly user is different from other users in one very important way. Many technology adoption models incorporate a phase which represents the stage where the person makes the decision to purchase the new innovation. Rogers [8] categorises people in terms of how long, after the product is launched, it takes for them to purchase the product. Hence, mobile phones are a special case because many of our participants did *not* purchase their mobile phones. Some were given the phones by children, others by friends.

For this group it seems that Rogers' [8] first three stages are skipped and the elderly user is often catapulted directly into the 4$^{th}$ stage: implementation, or the second stage of the domestication process, *objectification*, without being able to purchase the particular product that best suits his/her needs. It appears that intention to use is influenced primarily by external factors such as filial affection and safety and security concerns. From our observations it is clear that ease of use does not mediate intention to use – but it *does* influence actual use of the mobile phone. Many of our participants had phones, and pronounced themselves mobile phone users, but did not actually use the phone because of usability issues.

Hence the unavoidable conclusion is that current technology acceptance models only partially model the factors that influence the acceptance of the mobile phone by the elderly. *Value = f(usefulness,ease of use),* and each of these aspects plays a vital role in contributing towards the value of the phone. Moreover, ease of use prevents our users from utilising available functionality. Thus elderly users are getting far less than the optimal value out of their mobile phones. Another problem with existing technology acceptance models is that they model mainly the appropriation phase without due consideration of the other phases in the adoption process.

The contribution of this paper is to propose the acceptance matrix as a new tool for relating the technology *acceptance factors* that have emerged from quantitative research to the *adoption phases* identified through quantitative research. Therefore the paper contributes to integrating findings from different fields, namely MIS and Sociology. The two-dimensional adoption matrix can be useful in identifying and even predicting gaps in the technology adoption process. The challenges that face each generation or user group are different, but the influence of acceptance factors on adoption phases, as presented by the adoption matrix, will remain relevant. Future work will focus on developing a technology acceptance model suitable for modelling the context of the elderly mobile phone user.

## References


1. Mallenius, S., Rossi, M., Tuunainen, V.K.: Factors affecting the adoption and use of mobile devices and services by elderly people – results from a pilot study. In: 6th Annual Global Mobility Roundtable 2007, Los Angeles (2007)
2. Venkatesh, V., Morris, M.G., Davis, G.B., Davis, F.D.: User acceptance of information technology: Toward a unified view. MIS Quarterly 27(3), 425–478 (2003)
3. Lee, Y.S.: Older adults' user experiences with mobile phones: Identification of user clusters and user requirements, in Industrial and Systems Engineering. Doctoral dissertation at Virginia Polytechnic Institute and State University, Virginia (2007)
4. Van Biljon, J.A.: A Model for Representing the Motivational and Cultural Factors that Influence Mobile Phone Usage Variety, Doctoral dissertation at the School of Computing. University of South Africa, Pretoria (2007)
5. Haddon, L.: Domestication and Mobile Telephony, In Machines that Become Us: The Social Context of Personal Communication Technology, pp. 43–56. Transaction Publishers, New Brunswick (2003)
6. Davis, F.D.: Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. MIS Quarterly 13(3), 319–340 (1989)
7. Ling, R.: The diffusion of mobile telephony among Norwegian teens: A report from after the revolution. Telenor Research and Development (2001)
8. Rogers, E.M.: Diffusion of Innovations, 5th edn. The Free Press, New York (2003)
9. Silverstone, R., Haddon, L.: Design and the Domestication of Information and Communication Technologies: Technical Change and Everyday Life. In: Communication by Design: The Politics of Information and Communication Technologies, Oxford, pp. 44–74 (1996)

10. Malhotra, Y., Galletta, D.F.: Extending the Technology Acceptance Model to Account for Social Influence: Theoretical Bases and Empirical Validation. In: Proceedings of the Thirty-Second Annual Hawaii International Conference on System Sciences, vol. 1. IEEE, Los Alamitos (1999)
11. Pedersen, E.: Adoption of Mobile Internet Services: An Exploratory Study of Mobile Commerce Early Adopters. Journal of Organizational Computing and Electronic Commerce 15(3), 203–222 (2005)
12. Kwon, H.S., Chidambaram, L.: A test of the technology acceptance model: The case of cellular telephone adoption. In: Proceedings of the 33rd Hawaii International Conference on System Sciences, vol. 1, pp. 1–10. IEEE Computer Society, Hawaii (2000)
13. Sarker, S., Wells, J.D.: Understanding Mobile Handheld Device use and adoption. Communications of the ACM 46(12), 35–40 (2003)
14. Phillips, L., Sternthal, B.: Age differences in information processing: a perspective on the aged consumer. Journal of Marketing Research 14(2), 444–457 (1977)
15. Abascal, J., Civit, A.: Universal Access to Mobile Telephony as a way to enhance the Autonomy of Elderly People. In: WUAUC 2001 (2001)
16. Ziefle, M., Bay, S.: Mental Models of a Cellular Phone Menu. In: Brewster, S.A., Dunlop, M.D. (eds.) Mobile HCI 2004. LNCS, vol. 3160, pp. 25–37. Springer, Heidelberg (2004)
17. Osman, Z., Maguir, M., Tarkiainen, M.: Older Users' Requirements for Location Based Services and Mobile Phones. In: Chittaro, L. (ed.) Mobile HCI 2003. LNCS, vol. 2795, pp. 352–357. Springer, Heidelberg (2003)
18. Kurniawan, A.: Mobile Phone Design for Older Persons, Interactions, pp. 24–25 (2007)
19. Chismar, W.G., Wiley-Patton, S.: Does the Extended Technology Acceptance Model Apply to Physicians. In: Proceedings of the 36th Annual Hawaii International Conference on System Sciences (2003), doi:10.1109/HICSS.2003.1174354

# Visualising the Dynamics of Unfolding Interactions on Mobile Devices

Kristine Deray[1] and Simeon J. Simoff[1,2]

[1] University of Technology, Sydney, NSW 2007 Australia
kristine.deray@uts.edu.au
[2] University of Western Sydney, NSW 1797 Australia
s.simoff@uws.edu.au

**Abstract.** Human activities in various domains include interactions. In some areas, like healthcare, there is an opportunity to collect rich data about such interactions. The utilisation of such data in order to extract information about the dynamics of unfolding of interactions and visualising such information on mobile devices is the focus of this paper. The paper presents a methodology for designing visual representations about interactions for mobile devices. The shape and behaviour of the elements of such visualisations utilise elements of human movement. The paper presents an instance of such visualisation on mobile devices and its utilisation on case studies of interactions in healthcare.

## 1 Introduction

Many human endeavours are based on collaboration. In healthcare, practitioner and patient collaborate to achieve a way of improving patient's state of health. In participatory design, designer(s) and client collaborate to develop the design that best suits the client. Collaboration relies on interactions, hence, it will benefit if there is a way of monitoring, understanding and evaluating interactions that occur during it and feeding that evaluation back into the interaction process, in order to improve the process of collaboration and its outcomes.

Interactions constitute a process that unfolds. The way interactions unfold can tell us a lot about both the process and the outcomes of the interactions. Hence, in order to improve processes that rely on interactions, it is important to develop methods of encoding information about the way interactions unfold. Both humans and machines should be able to utilise such information during the interaction process in order to improve interactions and respective processes. For example, in health care, the treatment process is expected to benefit if we can present the information about how patient-practitioner interactions unfold, and if, based on that representation, one can judge about whether there has been a good communication or not, and if not, where were the bottlenecks [1]. The participatory design process may benefit if we can present the information about how client-designer interactions unfold and if, based on that representation, one can judge about whether design communication has been good or not, and use the interaction patterns to align with the emergence of design solutions, in order to revisit intermediate design solutions [2].

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 238–247, 2008.

In both examples the timely delivery of the information about interactions - the interaction profile, to the point of decision-making is vital. The capability of mobile devices to deliver information to the point of decision making makes them attractive for facilitation of decision making. The application areas span from the management in diverse business areas to the delivery of health care. The changes in the decision making processes, caused by these technological developments, reflect back on the requirements towards the designers of mobile devices in terms of offered functionality. These requirements face significant constraints, based on the size of the mobile devices, in this case in particular, on the size of display and interface controls.

Decision-making usually involves people with different backgrounds, who need to operate with a variety of information, presented at different levels of granularity. The goal of this research is the development of technology for visual representation of information about interactions that caters for people with diverse backgrounds, who are involved in decision making processes.

Visual representations translate data and information into a visible form that highlights important features, including commonalities and anomalies. Such visual representations make it easier for analysts and decision makers to grasp key aspects of presented information quickly [3].

The design of visual representations of information has been the focus of the fields of scientific data visualisation, information visualisation and design computing. All three fields create graphical models and visual representations from data that support direct user interaction. In the first two fields the aim is to explore large data sets and information spaces acquiring insights into useful information embedded there [4]. In scientific data visualisation graphical models are derived from measured or simulated data, which represent concepts associated with objects in the physical world, when in information visualisation, the graphical models may represent abstract concepts and relationships that do not necessarily have a counterpart in the physical world. In many visualisations, both approaches use concepts from a commonly known theme to express the semantics of domain data in popular terms. For instance, landscape is a popular theme for visualising diverse data sets, ranging from climate data sets to large collections of text documents (for example, high peaks in the IN-SPIRE™ document landscape visualisation show prominent topics, and peaks that are close together represent clusters of similar documents [3]). The third field - design computing, deals with the visualisation of physical objects, for example, building and interior design models, and abstract concepts, for example, spaces in virtual worlds. These visualisations aim to represent different aspects of designs, facilitating the development and evolution of design ideas [5].

All these areas have developed some schemes to visualise communication and interaction data. The tendency so far has been to present interactions in terms of the results they produce rather than to provide the mechanisms to encode and display "how" interactions unfold in time [1]. For example, network type of visualisations focus on displaying the relations between interacting entities. In [2] authors present an attempt to depict patterns in unfolding of collaboration within teams based on a nested-rectangles visual representation of on-line communication utterances between team members.

Creating well-constructed visual representations remains a challenge (see Chapter 3 in [3] for some of the issues facing designers of visual representations). So far creating visualisations has been essentially an adhoc process with very little or none formal design methods, engineering as a technology and consistent evaluation [4]. In [1] the authors presented a framework for designing visual languages based on human movement and proposed an example of a visual language. The language utilised the strength of humans in recognizing intuitively constructs in interactions modelled on human movement. In this paper we consider the design of the visualisation in the context of mobile computing and the delivery of the visual expressions on mobile devices.

Human movement provides us with the metaphor for modelling the domain of interactions through understandings derived from the intrinsic dimensions and qualities of the system of human movement. The argument for utilising human movement constructs places the body as central to understanding and deriving the connecting metaphor for the domain of interactions [6].

In this paper we present a methodology for designing the underlying representation and corresponding visualisations of the information about the dynamics of unfolding interactions. The method is based on elements of human movement. The paper specifies the requirements towards the visualisation of information about interactions on a mobile device. It presents an instance of such visualisation on mobile devices and demonstrates its utilisation in case studies of interactions in healthcare. The paper concludes with a summary and directions of future research.

## 2 Method for Designing Representations and Visualisations of Interactions

Proposed visualisation methodology is based on consistent semantic mapping and rigorous underlying formalisms. Fig. 1 presents compactly the conceptual modelling that enables the method for designing representations of interactions following the elements of human movement. The mapping, inspired by the Lakoff and Johnson's approach to metaphors [7], allows to express the target domain through the constructs of the source domain. In the process in Fig. 1 the source domain human movement (a) is interpreted and formalised through the methodology developed in Movement observation science (b) [8], deriving the constructs of *elasticities* and *qualities* (c). These two groups of constructs provide *shaping affinities* for expression of interaction (e). The target domain (d), the domain of interactions, is expressed through the shaping affinities represented in the behaviour of the elasticities and qualities (f). The patterns of interaction (g) between involved parties [$P_1$ and $P_2$ in Fig. 1] are expressed through the behaviour of these constructs. The mapping, illustrated by the process in Fig. 1, takes concepts that describe two frames of reference of human movement - (i) body position [the place of the body in space]; and (ii) body dynamics [the motion that causes and expresses change from one position of the body to another], and uses their computational representations in the interactions domain. The set of constructs considered in this paper include the following elasticities – the *rising and sinking* (RS-) elasticity and the *contraction and extension* (CE-) elasticity. Table 1 illustrates the linking of these key groups of constructs in the source and target domains, and briefly describes the way these concepts relate in the context of their function in the target domain. The behaviour of elasticities describes the reciprocal effects between parties in human movement interaction.

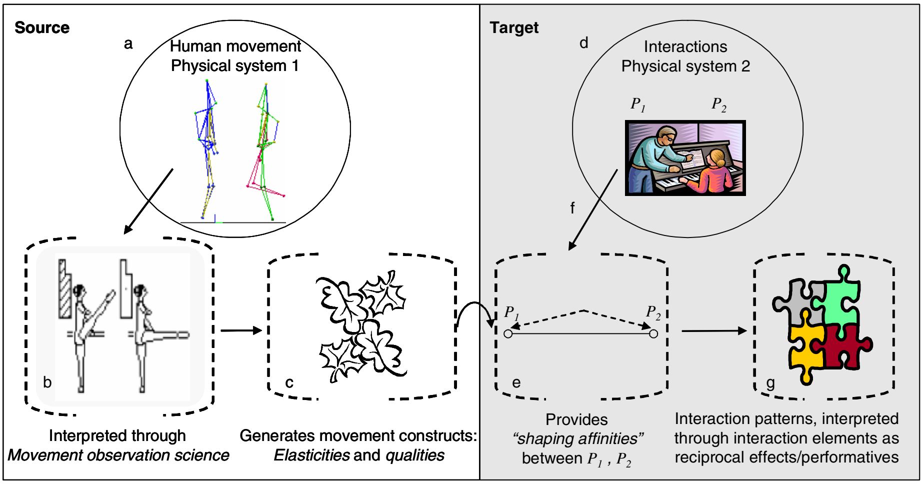



**Fig. 1.** Linking the domains of human movement and interactions through a mapping as two physical systems

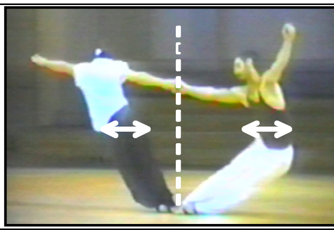
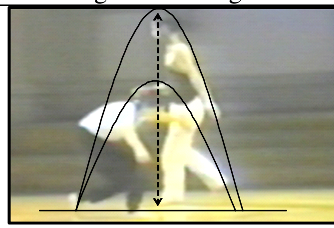
**Table 1.** An overview of the key groups of constructs derived from the domain of human movement and relating them to their function in the target domain of interaction representation

| Source domain constructs | Target domain constructs | Mapping between source and target domains |
|---|---|---|
| (a) Elasticities | (c) Reciprocal effects | Represent the reciprocal effects as integral dimensions of the domain of interaction. |
| (b) Qualities | (d) Performatives | Represent cooperative processes |

How the interaction functions, for instance, is it independent from other interactions, is it symmetrical or asymmetrical, balanced or unbalanced – all these are expressed by the changing shape and structure of the elasticities. Qualities generate a canonical structure in both domains. Similar to the use of the term in music, where a number of instruments or voices contribute to the overall structure with some modification in a dimension such as pitch, the basic idea is that parts combine to make the whole and that each part expresses a variable on a dimension that is integral to the domain. In the target domain this notion references the concept of "common ground" defined as an aggregated measure derived from the dynamics of participating parties in interactions.

As form and function are intrinsically coupled in human movement, on the one hand, and, the purpose of the visualisation system is to allow monitoring interactions and depict their patterns in terms of the meta-information extracted, we have adapted the "Form-Semantics-Function" methodology for development of visualisation techniques for visual data mining and analytics [9, 10]. Table 2 presents an example of the formal derivation of the form constraints for the RS- and CE-elasticities. The movement analysis has been performed over contact-improvisation data. The action of the

**Table 2.** Deriving the forms of the basic elasticities from the source domain based on the corresponding anatomical actions

| **Elasticity** | Contraction and Extension | Rising and Sinking |
|---|---|---|
| **Form** |  |  |
| **Semantics** | The amplitude on the horizontal axis indicates the strength of interaction. In terms of interpretation, it models strength as states of attraction, repulsion or stable behaviour. | The amplitude on the vertical axis indicates the elastcities that stretch up and sink down. |
| **Function** | Describes the intensity of the interaction. | Describes flow and effort of the interaction. |

two elasticities has been separated in vertical and horizontal directions, respectively, spanning a 2D representation of interactions.

Elasticities are descried through the computational models of their qualities. In its current development the RS-elasticity includes four qualities, presented here in terms of their interpretation in the interaction domain:

- *flow* ( $q_1^{R-S}$ ), which characterises the obstruction (e.g. language/social/ ultural, etc);
- *transition* ( $q_2^{R-S}$ ), which characterises interaction in time;
- *exertion* ( $q_3^{R-S}$ ), which correlates to the amount of effort required for an interaction to achieve some perceived position; and
- *control* ( $q_4^{R-S}$ ), which indicates the amount of control applied in the interaction.

The CE-elasticity currently includes a single quality – *intensity* ( $q_1^{C-E}$ ), which indicates the strength of the interaction.

Interactions are treated as a sequence of actions, where each action belongs to one of participants in the interaction. We demonstrate how qualities are computed by introducing a new quantitative representation of the flow. Flow ( $q_1^{R-S}$ ) is a measure of the contribution of each participant to an interaction. To measure flow we look to the difference between the lengths of each participant's actions in time. Flow is proportional to the difference between the relative values of the length of participant's actions. We consider that each participant contributes a number of actions to the interaction. Let consider an interaction *j* between two parties', $P_k$ and $P_l$ in the interaction *j*. Let $t_i(P_k)$, and $t_i(P_l)$, be the respective action times contributed by $P_k$ and $P_l$ to the interaction *j*. Let $n_k$ and $n_l$ be the number of actions in the interaction *j* for each participant, respectively. Then the flow, $Flow_j(P_k,P_l)$ of the interaction *j* is defined as

$$Flow_j(P_k,P_l) = \frac{1}{\max\{n_k,n_l\}} \sum_{i=1}^{\max\{(n_k,n_l\}} \left[1 - \left|\frac{t_i(P_k) - t_i(P_l)}{t_i(P_k) + t_i(P_l)}\right|\right] \quad (1)$$

The measure of flow, defined in equation (1) reflects domination in interaction by party with a large number of actions and greater duration of the actions than those contributed by the other party.

In a similar way, the computational models of the other qualities are derived. The details of these computational models are presented in [1], except for the flow, which is treated in [1] qualitatively. These representations are then utilised by the visualisation model.

### 2.1 Requirements towards the Visual Representation of Interactions for Mobile Devices

One of the goals of this research is visualising the information about the real-time dynamics of interactions on a static display. In [1] the authors have specified and addressed a set of requirements towards the design of the visual components (primitives and their aggregations) for constructing a visual language based on the elements of human movement. The set includes: symmetry, regularity, similarity, connectivity, legend, containment and aggregation, and transparency. In this paper, we extend this set with additional requirements in order for a visualisation to be suitable for mobile devices. These additional properties, which bear some similarity to the requirements towards the "parallel coordinates" visualisation [11], include complexity, expandability and scalability. "Complexity" aims at low representational complexity, so that through the visual representation the development and unfolding of interactions can be monitored in real-time on a mobile device. Expandability aims at visualisation that can accommodate change in the representation of interaction information. Scalability addresses not only the size of the data displayed, but the diversity of screens on which visualisation can appear, so that the displayed components remain recognisable under projective transformations, such as scaling, rotation, translation or if presented in perspective.

Fig. 2 presents the application of our approach, shown in Fig. 1 to the design of the visual representation of the qualities of respective elasticities, identified in Laban's system. The rules for constructing the visual elements, derived from Laban's system and addressing the Form-Semantics-Function properties described in Table 2, are summarised in Table 3.

The visual elements of the CE- and RS-elasticities operate in horizontal and vertical direction, respectively. The production element is the visualisation unit which represents an interaction segment. An interaction session is a collection of interaction

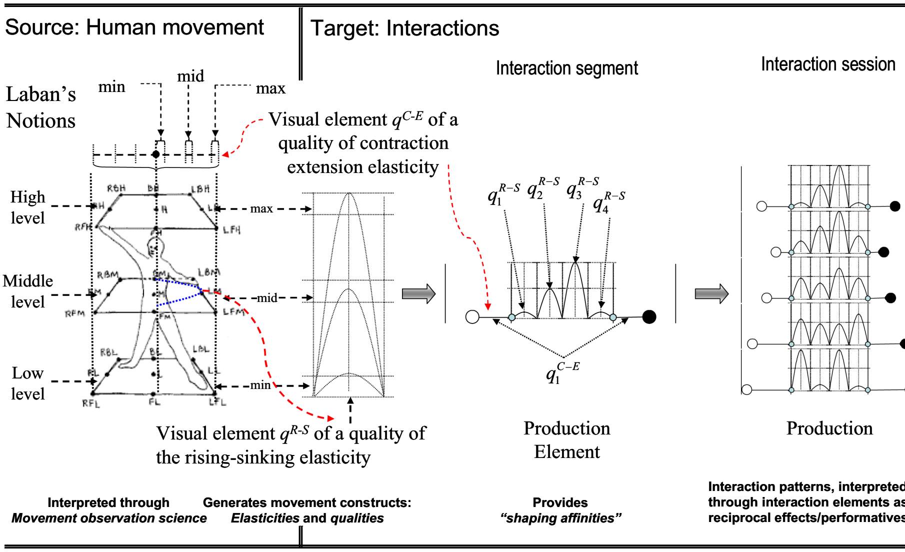



**Fig. 2.** The elements of human movement and their constructs, extracted from the movement observation science provide the basis for the defining the shape, behaviour and aggregation of the constructs of the visual representation of interactions

**Table 3.** Composition rules for the visual representations of the qualities of the elasticities

| Rule | Visual element of the CE-elasticity | Visual element of the RS-elasticity |
|---|---|---|
| 1 | Each arm of $q_i^{C-E}$ starts at the internal border of the track segment and expands towards the external border. | Each curve representing the value of $q_i^{R-S}$ is symmetric with respect to the vertical access passing through the middle of the cell. |
| 2 | The arms that represent the first quality $q_1^{C-E}$ are aligned with the lower end of the element; the anchor points are attached to the arms of $q_1^{C-E}$. | The cell that represents the first quality $q_1^{R-S}$ is aligned with the left end of the effort-shape element. |
| 3 | The arms that represent each subsequent quality $q_i^{C-E}$ are positioned above the previous arm pair; the correspondence of position to the index number does not change | The cells that represent each subsequent quality $q_i^{R-S}$ of the "Rising and Sinking" elasticity are positioned to the right of the previous cell; the correspondence of position to the index number does not change. |

segments. It is represented by a collection of production elements that constitutes a production. Productions are visual expressions. They provide rich and compact view of the different sets of interactions, allowing to (i) grasp the macro-picture of the interaction flow; (ii) perform comparative analysis of different parts of the same interaction; and (iii) perform comparative analysis of different interactions.

## 3 Interaction Visualisation on Mobile Devices in e-Health

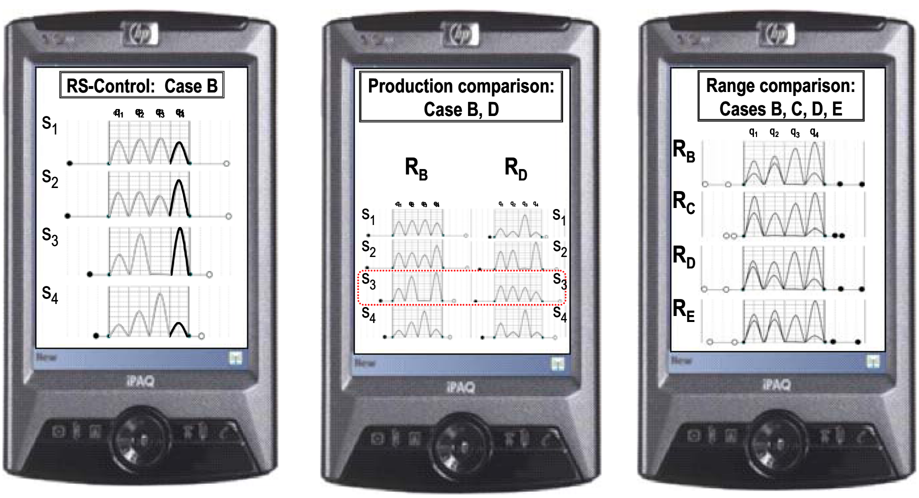
Interactions between medical practitioners and patients play a key role in healthcare, for instance, in determining the accuracy of diagnosis, patients' commitment to treatment regimes, and the extent to which patients are satisfied with the service they receive. The importance of interactions has been recognised with the development of relationship-centred healthcare (see [12] for a comprehensive literature review). Such interactions may involve parties with different background. Fig. 3 and Fig. 4 present different information about the interaction profiles of patient-practitioner sessions and how they are integrated through the mobile devices in the e-health process. Fig. 3 presents an example from the domain of occupational therapy, labelled as Case B. It deals with the suitability of an elderly patient for discharge from hospital to her own home where she lives alone. The interaction follows the assessment of the patient by obtaining some measure(s) of "fitness". The patient and practitioner interact through actions that are verbal and non-verbal. Each action has a time length, which is an argument in the functions that compute the values of the different qualities.

Fig. 3a presents an example of a production element. The inclusion of the video is optional, subject to the legal agreements. Based on the values, an indicative interpretation about the interaction in terms of the elasticities can be provided. The production view in Fig. 3b shows the entire interaction in Case B, which included four segments, labelled as $s_1$, … $s_4$. In the production the variation of the shape of: flow ($q_1$) indicates an average communication between the parties; transition ($q_2$) indicates that communication between parties occurred with reasonably quick responses to actions of fairly short lengths; exertion ($q_3$) indicates complex level of interaction as it shows greater effort, especially in the last segment; control ($q_4$) shows some flexibility in the middle of the session (segments 2 and 3) with a dominance of the practitioner towards the end. Fig. 3c shows the production range element, which indicates the ranges for each of the qualities during the session.

Fig. 4 shows a sample of views for performing different comparisons. Fig. 4a shows a comparison along a single quality – the control quality in Case B. In general, comparison of one or more qualities, switching the rest of them in background (context) mode, is a visual analytics technique that can be applied if we want to emphasise the dynamics of one or more specific qualities. Fig. 4b shows an example of comparison of productions representing the interactions of two different cases. Case D is part of the examination of the interaction between a doctor and a patient during a post-operative consultation. The outcome of the task is a diagnosis that varies dependent on the interaction between the two parties. This fragment is snapshot of a poor communication, where clinician does not wish to listen to the patient story. This mode offers comparison both of complete productions and of pairs of interactions. n Fig. 4b, the comparison of interaction segments $S_3$ shows that in Case B in qualities of effort shape perform with irregular profile with exertion at min and control at max, indicating a clearly defined drive of the interaction. Case D shows an interaction with a substantial control. Comparisons across cases can include comparison between range elements as shown in Fig. 4c.


a. segment view b. production view c. range view

**Fig. 3.** Examples of views of the information about interactions that is available to the parties




a. $q_4$-slice comparison b. production comparison c. range comparison

**Fig. 4.** Examples of different comparison modes

## 4 Conclusions

Interactions play central role in our technologically enriched and enabled life. When these interactions are facilitated by information technology we can collect detail data about them and incorporate such data in the decision making process. This paper presented a methodology for developing visual presentations of the information about interactions on mobile devices, in order to utilise such intelligence in subsequent activities. The shape and behaviour of the visual representations are based on elements of human movement as such elements are encoded in our common knowledge [6]. The paper formulated the requirements for such visualisations and demonstrated an instance of visualisation of information about interaction dynamics for mobile devices that addresses this need and complies with the requirements towards such visualisations. Proposed visualisation and supporting operations make complex dynamics of interactions compactly visualised on mobile devices. It enables exploration, comparison, summarisation, quality-specific analysis, tracking of particular patterns, and many other operations over the profiles of interactions.

**Acknowledgements.** This research is supported by University of Technology, Sydney and University of Western Sydney.

## References

1. Deray, K., Simoff, S.J.: Designing a visual language for interaction representation based on aspects of human movement. In: Ferri, F. (ed.) Visual Languages for Interactive Computing: Definitions and Formalizations, pp. 205–231. IGI Global (formerly Idea Group Inc.), Hershey (2007)
2. Simoff, S.J., Maher, M.L.: Analysing participation in collaborative design environments. Design studies 21, 119–144 (2000)
3. Thomas, J.J., Cook, K.A.: Illuminating the Path: The Research and Development Agenda for Visual Analytics. IEEE CS Press, Los Alamitos (2005)
4. Ferreira de Oliveira, M.C., Levkowitz, H.: From Visual Data Exploration to Visual Data Mining: A Survey. IEEE Transactions On Visualization And Computer Graphics 9(3), 378–394 (2003)
5. Maher, M.L., Simoff, S.J., Cicognani, A.: Understanding virtual design studios. Springer, London (2000)
6. Pheifer, R., Bongard, J.: How the Body Shapes the Way We Think. MIT Press, Cambridge (2007)
7. Lakoff, G., Johnson, M.: Metaphors We Live By. University of Chicago Press, Chicago (1980)
8. Newlove, J., Dalby, J.: Laban for All. Nick Hern Publishers, London (2004)
9. Simoff, S.J.: Towards the development of environments for designing visualisation support for visual data mining. In: Proceedings International Workshop on Visual Data Mining VDM@PKDD 2001, Freiburg, Germany (2001)
10. Simoff, S.J.: Form-Semantics-Function - A framework for designing visual data representations for visual data mining. In: Simoff, S.J., Böhlen, M.H., Mazeika, A. (eds.) Visual Data Mining: Theory, Techniques and Tools for Visual Analytics, pp. 30–45. Springer, Heidelberg (2008)
11. Inselberg, A.: Multidimensional detective. In: Proceedings of the IEEE Symposium on Information Visualization. IEEE Press, Los Alamitos (1997)
12. Wylie, J.L., Wagenfeld-Heintz, E.: Development of relationship-centered care. Journal for Healthcare Quality 26(1), 14–21 (2004)

# Preface to RIGiM 2008

Colette Rolland[1], Carson Woo[2], and Camille Salinesi[1]

[1] Université Paris 1 Panthéon - Sorbonne, France
[2] University of British Columbia, Canada

The use of intentional concepts, the notion of "goal" in particular, has been prominent in recent approaches to requirement engineering (RE). Goal-oriented frameworks and methods for requirements engineering (GORE) have been keynote topics in requirements engineering, conceptual modelling, and more generally in software engineering. What are the conceptual modelling foundations in these approaches? RIGiM (Requirements Intentions and Goals in Conceptual Modelling) aims to provide a forum for discussing the interplay between requirements engineering and conceptual modelling, and in particular, to investigate how goal- and intention-driven approaches help in conceptualising purposeful systems. What are the fundamental objectives and premises of requirements engineering and conceptual modelling respectively, and how can they complement each other? What are the demands on conceptual modelling from the standpoint of requirements engineering? What conceptual modelling techniques can be further taken advantage of in requirements engineering? What are the upcoming modelling challenges and issues in GORE? What are the unresolved open questions? What lessons are there to be learnt from industrial experiences? What empirical data are there to support the cost-benefit analysis when adopting GORE methods? Are there applications domains or types of project settings for which goals and intentional approaches are particularly suitable or not suitable? What degree of formalization and automation or interactivity are feasible and appropriate for what types of participants during requirements engineering? e.g., business domain stakeholders, requirements modellers, ontology engineers, etc.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, p. 248. 2008.

# Reflective Analysis of the Syntax and Semantics of the i* Framework

Jennifer Horkoff[1], Golnaz Elahi[1], Samer Abdulhadi[2], and Eric Yu[2]

[1] Department of Computer Science, University of Toronto, Canada, M5S 1A4
{jenhork,gelahi}@cs.toronto.edu
[2] Faculty of Information, University of Toronto, Canada, M5S 3G6
yu@fis.utoronto.edu, samer.abdulhadi@utoronto.ca

**Abstract.** Conceptual modeling notations are often designed without the benefit of empirical input. Reflective analysis of modeling languages can help find the gap between the intended design of the language and its use in practice. In this paper, we study instances of the i* goal and agent-oriented Framework to analyze differences between the core i* syntax developed at the University of Toronto and existing variations. We have surveyed 15 student assignments and 15 academic papers and presentations in order to capture and analyze the most common i* syntax variations. Through this analysis we offer insights into i* syntax and suggestions to improve the framework and increase consistency between models.

**Keywords:** Goal Orientation, i* Modeling Framework, Language Variation.

## 1 Introduction

In system development, modeling languages are introduced to serve various purposes, such as facilitating communication, making implicit information explicit, or acting as a repository for knowledge. As a modeling language is adopted and used, it is useful to actively reflect on the syntax and semantics of the language, including the original intention of the language and how it is actually used in practice, in order to inform the community of users and make refinements to the language.

In this study, we focus on the i* Conceptual Modeling Framework, a goal and agent-oriented framework introduced in [1]. The Framework is intended to capture the desires which motivate system development from an agent-oriented point of view, capturing the goals of and interactions amongst system stakeholders and the system. It is directed towards the discovery and comparison of alternative system designs at a high level of abstraction, including the assignment of alternative responsibilities and the exploration of different goal operationalizations.

Although suggestions and potential directions for use of the modeling notation have been provided [1] [2], the description of the Framework was left open to a certain degree of interpretation and adaptation. Consequently, the framework has been applied to many different areas, including requirements engineering [2], system design methodologies [3] [4], and security analysis [13] [14].

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 249–260, 2008.

In existing work using i*, the core syntax of the i* framework has often been modified, or has evolved in different directions. Furthermore, experiences teaching the i* Framework to students has shown that they often modify the syntax presented to them, either intentionally or unintentionally. In this study, we survey various instantiations of i* models and compare these instances to the version of i* currently used at the University of Toronto (U of T) [2]. A reflective analysis of the i* Framework is performed, looking critically at our own use of the framework, and questioning the assumptions underlying syntactic choices.

In this work, we call differences between this and other styles of modeling variations. We are interested in discovering the most commonly occurred variations for both students, learning i*, and researchers, applying i* in their work. To this end, 15 student assignments and 15 academic works containing examples of i* models have been surveyed. A qualitative analysis has been performed in order to understand the motivations behind the syntax variations. In our analysis, we compare the perceived motivations behind these variations to the original motivations behind the U of T syntax. As a result of our analysis, we clarify the meaning of several commonly occurring syntactical structures, make recommendations for the use of i* syntax, and provide some suggestions for potential language modifications.

## 2 Related Work

Although use of graphical models and modeling processes have been widely studied in software engineering research, the design of modeling notations is largely unscientific [5]. For example, the decisions about how to graphically represent constructs are made based on personal taste and intuition rather than scientific evidence [5]. In addition, Gurr and Tourlas [6] assert that many formal accounts of diagrammatic languages confuse or destroy any natural reading of the diagrams. Moody [5] argues that most diagrams do not communicate effectively and act as a barrier rather than an aid.

State of the art research in modeling languages raises the question of how modeling languages should be designed systematically, and how the existing modeling notation should evolve methodologically? The contribution in [6] expresses the need to define intuitive and natural models to move toward principled design of modeling notations. Gurr and Tourlas assert that an “intuitive” representation is one which is well matched to what it represents. Furthermore, whether a representation is “natural” concerns how it achieves its intuitive matching.

A few research studies address common mistakes specifically in using modeling notations. Among them, Lange and Chaudron [7] report results from observations in industry that show the amount of violations of completeness and consistency rules among UML models developed in practice is very high. They also conducted an empirical study to investigate to what extent implementers detect defects and to what extent defects cause different interpretations by different readers [8].

Although defect detection is not a concern of our work, we are nevertheless interested in the effect of syntax variances on understandability and comprehension of the models. We also differ from [8] in our approach to the study. In [8], UML models with deliberate defects were presented to the subjects of the experiment; while in our approach, we overviewed students' course assignments and academic papers to discover common mistakes.

Finally, Webster et al. [9] survey several published papers to collect both good practices and misuses of the i* Framework, to help requirements engineers to use the i* Framework in its full capacity. The misuses of i* mentioned in [9] overlap with the common variations we discover in this work. However, we further analyze the mistakes to discover why users make the variations and how the modeling language could be refined according to the current practice.

To our current knowledge, no studies exist which gather common syntactical variations of a modeling notation and suggests refinements to the modeling language based on patterns of mistakes that users make in practice.

## 3 The i* Framework

Here, we present the i* syntax initially developed at University of Toronto [1], and evolved in work such as [2], as the baseline for the comparison of surveyed models. Focusing on this syntax does not imply that it is the default or primary syntax, in fact we aim to analyze the common variations of this syntax in order to understand its weaknesses and improve its use. The i* modeling notation includes elements such as goals, softgoals, tasks, resources, and relationship links such as decomposition, dependencies, means-ends, and contributions. The Framework consists of two main modeling components: the Strategic Dependency (SD) model and the Strategic Rationale (SR) model.

In an SD model, actors in an organizational setting are defined in terms of positions, roles, and agents. The i* modeling notation provides association links to relate actors. In addition, actors can depend on each other to achieve a goals or softgoal, perform a task, or furnish a resource. An SR model extends an SD by providing task decomposition, goal refinement, and goals contributions inside the boundaries of actors. *Task-decomposition links* provide a hierarchical description of intentional elements which accomplish a task. The *means-ends links* break down a goal into alternative tasks that achieve the goal. *Contribution links* such as *help*, *make*, *hurt*, and *break* are used to express the impact of an element on softgoals. According to this syntax, tasks can be decomposed to any element. Goals can be refined using means-end relationship, and all the elements can contribute to softgoals.

Models in i* provide a basis for goal model evaluation. Procedures such as the one described in [10] propagate the level of satisfaction and denial of elements, based on a selection of alternatives, to ensure that actors' top level goals are satisfied.

## 4 Survey Method and Results

In order to enhance the consistency and effectiveness of the modeling process, and provide a resource for students, a set of guidelines based on the U of T i* syntax was developed and added to the i* Guide on the i* Wiki [11]. Guidelines were inspired by finding common variations via a survey of previous assignments in graduate systems analysis courses, as well as a set of available publications, all using i*. Inspired by the findings of this unofficial survey, 15 essay-type student assignments from 2006 and 2007 and 15 academic papers and presentations were surveyed again, this time taking counts of specific variations to discover the most frequent variations. The assignments were taken from three semesters of graduate courses taught by one of the authors. All assignments were surveyed, unless the students expressed an unwillingness to participate. The papers and presentations were drawn from an introductory roadmap of i* publications selected by Yu, work appearing in the i* Workshops, and work listed in the i* Wiki (see the list of Surveyed Work in the Reference Section). The survey covered all constructs of all models in each paper and assignment. The domains covered by the surveyed models were diverse, including health care, banking, and education systems. The inclusion of both academic works and student assignments, allowed for a comparison of the types of variations made by students and newcomers to the variations made in research.

In the surveyed assignments, papers, and presentations, when a model was developed using a convention contrary to the i* guidelines, it was recorded as a variation. If a variation occurred several times in one source, it was only counted once, thus the totals represent the number of assignments or academic works in which the variation occurred. Some variations, such as the use of certain links with certain elements, were clear-cut to identify, while other variations involved a certain degree of subjective judgment in their identification, for example, deciding that a softgoal should be a goal. In addition to analyzing the variations in 30 sources, we performed a qualitative analysis of the motivations behind the variations, asking questions such as: What did the modeler mean to model with this variation? Is the underlying meaning clear? Was the variation deliberate? Why was the modeler driven to make this variation?

Table 1 lists common variations in the surveyed assignments and papers. It provides total number of variations for each category of variations for student assignments and academic papers or presentations. Although we analyzed both SD and SR models, we detected variations only in SR models. Several of the variations are explained in more detail in Section 5.

## 5 Analysis

By performing a qualitative analysis of the motivations behind each variation, we can analyze whether the variation represents a potential source of confusion in a model, whether the convention can be seen as a shortcut for more complex syntax, or whether it indicates issues in the i* notation used at U of T. We have grouped variations together when we believe they stem from similar motivations. The counts for each variation within a particular category are provided. Due to space restrictions, we pick out only the most prominent variations to discuss.

**Table 1.** Summary of common variations

| Category | Variations and grouping categories | Total # of instances for Assignments | Total # of instances for Papers/ Presentations | Total # of instances per variations |
|---|---|---|---|---|
| Decomposition Links | Decomposition links are drawn directly from goals to tasks | 5 | 4 | 9 |
| | Decomposition links are used between goals | 4 | 2 | 6 |
| | Decomposition links are drawn from goals to softgoals | 2 | 3 | 5 |
| | Decomposition links extend outside actors' boundaries | 1 | 3 | 4 |
| | Decomposition links are used between Softgoals | 2 | 1 | 3 |
| | Decomposition links drawn from softgoals to tasks | 2 | 0 | 2 |
| | Decomposition links are used between resources | 1 | 0 | 1 |
| | Decomposition links are drawn from goals to resources | 0 | 1 | 1 |
| Dependency Links | Dependency links are used in more than one strategic relationship | 4 | 4 | 8 |
| | Softgoal dependency is met by a goal | 5 | 0 | 5 |
| | Softgoal dependency is met by a task | 1 | 1 | 2 |
| | Dependency links are used inside actors | 0 | 1 | 1 |
| | Dependency links do not have dependums | 0 | 1 | 1 |
| | Dependencies link to actor boundary | 0 | 1 | 1 |
| Means-Ends Links | Means-Ends links are used between tasks | 2 | 1 | 3 |
| | Means-Ends links are used between goals | 1 | 2 | 3 |
| | Means-Ends extend outside actors' boundaries | 0 | 3 | 3 |
| | Means-Ends are drawn from goals to softgoals | 2 | 0 | 2 |
| | Means-Ends are drawn from goals to tasks | 1 | 1 | 2 |
| | Means-Ends are drawn from softgoals to goals | 1 | 1 | 2 |
| | Mean-Ends are used between softgoals | 1 | 0 | 1 |
| | Means-Ends are drawn from resources to goals | 0 | 1 | 1 |
| Contribution Links | Contribution links extend outside actors' boundaries | 1 | 5 | 6 |
| | Contribution links are drawn from softgoals to tasks | 3 | 1 | 4 |
| | Contribution links are drawn from Softgoals to goals | 1 | 1 | 2 |
| | Contribution links are used between goals | 1 | 0 | 1 |
| | Contribution links are drawn from resources to tasks | 1 | 0 | 1 |
| Element Types | Softgoal should be goal | 10 | 0 | 10 |
| | Goal should be softgoal | 11 | 4 | 15 |
| | Task should be softgoal | 7 | 1 | 8 |
| | Softgoal should be task | 7 | 0 | 7 |
| Other | Association links are used between incorrect specialized actors | 5 | 0 | 5 |
| | Softgoals are not decomposed | 2 | 0 | 2 |
| | Actors are included inside another actor | 0 | 1 | 1 |
| | Evaluation Labels are not propagated throughout the model | 0 | 1 | 1 |
| | **Totals** | **84** | **45** | **129** |

**The Nature of "Hard" Elements and Softgoals.** [Decomposition links are used between Softgoals (3), Decomposition links drawn from softgoals to tasks (2), Means-Ends are drawn from goals to softgoals (2), and Mean-Ends are used between softgoals (1), Softgoal dependency is met by a goal (5), Softgoal dependency is met by a task (2), Contribution links are drawn from softgoals to tasks (4), Contribution links are drawn from Softgoals to goals (2), Contribution links are used between goals (1), Contribution links are drawn from resources to tasks (1), Decomposition links are drawn from goals to softgoals (5), Means-Ends are drawn from softgoals to goals (2), Softgoal should be goal (10), Goal should be softgoal (15), Task should be softgoal (8), Softgoal should be task (7), Total: 70].

Many of the variations can be attributed to a misunderstanding of the nature of hard and softgoals. Generally, users confuse hard and soft elements. Furthermore, several variations involved having a "hard", non-softgoal element as a recipient of contribution links. In i*, a goal, task or resource is typically considered similarly to a functional requirement, they are concrete states, actions or entities, respectively. From this point of view, it does not make sense to say that another element can provide a qualitative contribution to these elements (either partial or sufficient). To keep the differences between hard and soft elements clear, we decompose hard elements using only AND/OR type links (Decomposition and Means-Ends) in order to ascribe clearly defined decompositions to concrete elements. We can also see that modelers occasionally use links associated with hard elements with softgoals, and that softgoals depend on hard elements in dependencies. For the first case, as the nature of a softgoal implies qualitative, "good enough" analysis, it is unlikely to be decomposable into strict AND or OR relationships, such as Means-Ends or Decomposition. Although the i* Framework does retain the use of AND and OR contribution links for softgoals, (adopted from the NFR framework [12]), their use is infrequent..

Similarly, when a softgoal dependency is met by a hard element, this may indicate a problem with the understanding of softgoals. In this situation, if the functional element (hard element) is satisfied, the qualitative aspect will also be satisfied. In some cases, the underlying meaning of this type of syntax may be desirable, similar to the situation where a Make link is used from hard element to a softgoal. However, if the contribution is only partial, or not positive, this syntax should be avoided.

In several cases, modelers have decomposed goals to softgoals, violating the restrictions that goals should only be decomposed to tasks. The nature of hard goals and softgoals implies that a softgoal should not be a means to a hard goal; sufficiently accomplishing a qualitative goal, should not allow the accomplishment of a concretely defined state of the domain. However, we can observe that in i* syntax a softgoal *is* allowed to be a decomposition element of a task. This seems to contradict the notion of a task as a concrete series of actions. In fact, when this type of syntax is used, we interpret the task to represent not only the concrete actions, but also the desired qualities that this particular task should accomplish in order to be satisfied. For example, in the left snippet of Fig. 1, Send Message is only satisfied if the Message is Sent Securely. If the message is sent, but it was not secure, send message is denied. Such a situation can also be created when tasks or goals depend on softgoals.

This situation may lead to potential confusion if a modeler or a model reader is not aware of this interpretation, and instead interprets Send Message as the binary,

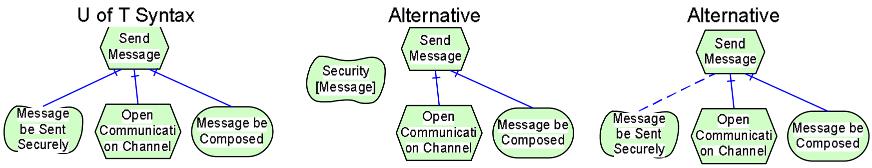



**Fig. 1.** Example Task Decomposition (left), Alternative Syntax Examples (middle, right)

concrete act of sending a message, where, even if the message is not sent securely, it can still be sent. Furthermore, as Send Message becomes a decomposition element of other functional elements, this implied qualitative aspect is passed up the decomposition tree to other elements which could be interpreted as entirely functional. In addition, if a task can be decomposed to a softgoal, why can a goal not also be decomposed in some way?

***Possible Responses.*** Although a solution to these issues may be to discontinue the decomposition of Tasks to softgoals, there remains a need to explicitly associate non-functional qualities with functional elements. In the NFR Framework [ ], this was done using a type and topic style of naming, where goals were named by the type of softgoal (security, ease of use, etc.) and their domain specific topic, as "Type [Topic]", see the middle of Fig. 1. Alternatively, a visual way to associate softgoals to functional elements which does not directly affect the evaluation of the functional elements could be devised, allowing the "hard" elements would retain their binary meaning. This alternative is shown on the right of Fig. 1. More investigation into the usability of the last option is required.

**Means-Ends vs. Decomposition.** [Decomposition links are drawn directly from goals to tasks (9), Decomposition links are used between goals (6), Means-Ends links are used between tasks (3), Means-Ends links are used between goals (3), Decomposition links are drawn from goals to resources (1), Means-Ends are drawn from goals to tasks (2), Total: 24]

In the U of T style of i* syntax, deliberate restrictions have been placed on the use of Decomposition and Means-Ends links between elements. A Decomposition link (AND Decomposition) is intended to be used only to decompose tasks into a combination of any element types, where as a Means-Ends link (OR Decomposition) is intended to be used only to refine a goal into alternative tasks. The results show that many i* users either chose to ignore or misunderstand these restrictions.

The restrictions concerning Decomposition and Means-Ends links can be justified by the notion of tasks versus goals, and by the desire to prompt for the discovery of alternatives. In [2], a goal, by definition, can be accomplished in different ways, whereas a task specifies one particular way of accomplishing something. Thus, in Fig. 2, modeling Appointment Be Scheduled as a Goal would indicate that there are several different ways to schedule an appointment, while choosing to model Financial Management as a Task indicates that this refers to one particular way of performing financial management.

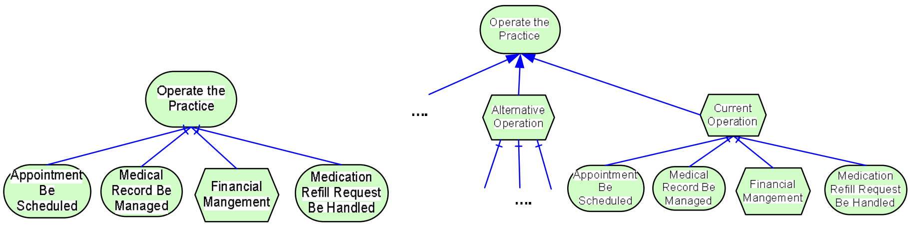



**Fig. 2.** Example of Decomposition Variation Recreated from [15] (left), Redrawn in the U of T Style (right)

In the U of T style, the left side of Fig. 2 would be redrawn as shown on the right. By adding the extra task (Current Operation) between the goal decompositions and the original goal, we emphasize that this set of decompositions composes only a single way to decompose and accomplish the task. There can, in fact, be several ways to decompose the high-level goal, each having potentially different effects on qualitative aspects represented as softgoals. Despite these reasons, for reasons relating to the scalability and simplicity of i* models, users often relax the rules concerning Means-Ends and Decompositions.

***Possible responses.*** With this in mind, we propose two levels of i* syntax, a strict level which follows the syntax laid out in Section 3 and a looser level which uses syntactical shortcuts. In the strict version of the syntax, restrictions such as those concerning Means-Ends and Decomposition would apply. In the looser level, these restrictions can be relaxed, allowing users to be more concise. Therefore we can consider the left side of Fig. 2 as a "shortcut" for the right side. When a modeler chooses to use this simplified syntax, the underlying meaning, represented by the more detailed syntax, should be clear to the modeler and the model readers. If there is any doubt concerning the clarity of meaning, the stricter syntax should be used.

**Actor Boundaries.** [Decomposition links extend outside actors' boundaries (4), Dependency links are used inside actors (1), Means-Ends extend outside actors' boundaries (3), and Contribution links extend outside actors' boundaries (6), Total: 14]

One frequently observed variation is that a decomposition, means-end, or contribution link extends outside of an actor's boundary. In the U of T version of i* syntax, all of these instances would be replaced by dependency links. It is important to limit non-dependency links to inside boundary of the actors to emphasize on actors' autonomy. In this way, externally visible actor relationships are limited to dependency link, and other actors do no have knowledge of the inside motivations of an actor. This situation better reflects the autonomy of actors occurring in the domain.

By only using dependency links across actor boundaries, one can ensure that the SR model is consistent with the SD model, avoid confusion translated between the two. However, practitioners frequently violate these rules, and according to detail analysis of the models, scalability and usability issues lead to these variations. Although these variations are not compatible with the notion of actor autonomy, they

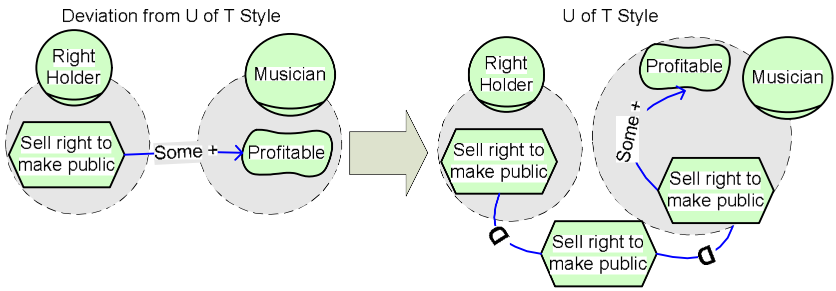



**Fig. 3.** An example of a variation used in a paper and its representation using the U of T style

communicate the same semantics represented with strict rules with a looser syntax which works as a shortcut. For example, Fig. 3 shows an example of a variation in [14] and its representation using the strict syntax of U of T style.

## 6 Discussion

By analyzing the results in Table 1, we can perceive several differences between the variations found in student assignments and academic work. It appears that students have more difficulty in understanding the nature of softgoals, and the differences between soft and hard elements. Although these notions are likely familiar to researchers, students are likely to be new to these ideas. Similarly, we see that students are more likely to have incomplete models, lacking softgoal decomposition, and are more likely to misuse association links. These issues can be addressed by placing greater emphasis on these concepts when teaching i* to new users.

On the other hand, we can observe that researchers are more likely to use non-dependency links outside of actor boundaries. We can postulate that researchers are more likely to adapt the Framework as they see fit. If they are faced with scalability issues, are more likely to deviate from the syntax laid out in [1] and [2]. No other significant differences between student and research results are found.

We can consider several threats to the validity of this study. First, the selection of academic papers and presentations was not performed in a completely random manner, and the surveyor was less interested in papers which did not have deviations. Therefore the selection is not necessarily representative of all research applying i*. However, the presence of a variety of domains in the research papers and assignments indicates that the discovered trends generalize across modeling subject matters.

When analyzing the differences between students and researchers, we see that the student assignments were often longer than the academic works, had more i* examples, and therefore had a higher chance of containing variations. However, as our observations of higher numbers of variations for student assignment is not universal across all counts, the trends observed likely remain valid.

Finally, the qualitative analysis of the variations found in both types of work was performed by the authors of this work, all of whom are very familiar with the U of T style of i* syntax, and who are biased by the flavor of i* which we have learned and used. Therefore, it is possible that the intention behind variations were misinterpreted in some cases. However, if the semantic intention of syntax variations can be misinterpreted, they may be ambiguous and problematic in general.

## 7 Future Work and Conclusions

Through a survey of student assignments and academic work, we have discovered frequent variations from the i* syntax developed at U of T. Through a qualitative analysis of these variations, we have attempted to understand the motivations for these variations, and compared these motivations to the original motivations for the syntax introduced by Yu [1] [2]. We suggest the use of strict and loose versions of i* syntax, the latter containing syntactical shortcuts for configurations in the former. The emphasis of this paper is not to avoid the loose syntax, but to be sure that modelers and model readers have a consistent interpretation of syntactical shortcuts. Suggestions for modifications to i* syntax involving a clearer separation of hard and soft elements have been made. Future work should involve empirical assessments of the effect of adopting the loose syntax on the resulting models and model reasoning.

The understanding facilitated by our analysis can promote consistency in i* modeling styles and model interpretation, and help to avoid ambiguous syntax. We have also pointed out potential areas of confusion for new i* users, allowing for a more effective formulation of curriculum involving i* modeling.

In the future, we would like to expand our survey to include more student assignments and academic papers in an attempt to verify or refute our results. In ongoing work, we have obtained access to i* models developed for industrial purposes in a software maintenance organization and are in the process of incorporating analysis of those models into the current study. We are also interested in examining the relationship between variations and model size, exploring to what degree variations are motivated by scalability issues. In addition, we are initiating a call to collect i* syntax variations from research groups using i* in the i* collaborative wiki. We have an "Open" version of the i* Guide, and invite researchers to contribute their varying i* syntax to the Guide [11]. By inviting others to provide examples and explanations of their syntax, we avoid the bias introduced when one group surveys the conventions of other groups. By collecting and comparing differing i* syntax, we can open a dialog concerning the motivations behind these differences, leading towards a more universal understanding of i* models.

**Acknowledgments.** We gratefully acknowledge financial support from the Natural Science and Engineering Research Council of Canada (NSERC) and Bell University Labs.

## References

1. Yu, E.: Modeling Strategic Relationships for Process Reengineering. Ph.D thesis, Department of Computer Science, University of Toronto, Canada (1995)
2. Yu, E.: Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. In: Proc. of the 3rd Requirements Engineering Conf., RE 1997, pp. 226–235 (1997)

3. Castro, J., Kolp, M., Mylopoulos, J.: A requirements-driven development methodology. In: Dittrich, K.R., Geppert, A., Norrie, M.C. (eds.) CAiSE 2001. LNCS, vol. 2068, pp. 108–123. Springer, Heidelberg (2001)
4. Jones, S., Maiden, N.A.M.: RESCUE: An integrated method for specifying requirements for complex socio-technical systems. In: Mate, J.L., Silva, A. (eds.) Requirements Engineering for Sociotechnical Systems. Idea Group Inc. (2005)
5. Moody, D.: What Makes a Good Diagram? Improving the Cognitive Effectiveness of Diagrams in IS Development. In: Proc. of the 15th Int. Conf. in Information Systems Development, ISD 2006, pp. 481–492 (2006)
6. Gurr, C., Tourlas, K.: Towards the principled design of software engineering diagrams. In: Proc. of the 22nd Int. Con. on Software Engineering, ICSE 2000, pp. 509–518 (2000)
7. Lange, C.F.J., Chaudron, M.R.V.: An Empirical Assessment of Completeness in UML Designs. In: Proc. of the Int. Conf. on Empirical Assessment in Software Engineering, EASE 2004, pp. 111–119 (2004)
8. Lange, C.F.J., Chaudron, M.R.V.: Effects of defects in UML models: an experimental investigation. In: Proc. of the 28th Int. Con. on Software Engineering, ICSE 2006, pp. 401–411 (2006)
9. Webster, I., Amaral, J., Cysneiros Filho, L.M.: A Survey of Good Practices and Misuses for Modelling with i* Framework. In: Proc. of the VIII Workshop on Requirements Engineering, WER 2005, pp. 148–160 (2005)
10. Horkoff, J.: Using i* Models for Evaluation, Masters Thesis, University of Toronto, Department of Computer Science (2006)
11. i* Guidelines, version 3 (2007), http://istar.rwth-aachen.de/tiki-index.php?page_ref_id=67
12. Chung, L., Nixon, B.A., Yu, E., Mylopoulos, J.: Non-Functional Requirements in Software Engineering. Kluwer Academic Publishing, Dordrecht (2000)
13. Liu, L., Yu, E., Mylopoulos, J.: Security and Privacy Requirements Analysis within a Social Setting. In: IEEE Joint Int. Conf. on Requirements Engineering, RE 2003, pp. 151–161 (2003)
14. Gans, G., Jarke, M., Lakemeyer, G., Schmitz, D.: Deliberation in a metadata-based modeling and simulation environment for inter-organizational networks. In: Eder, J., Missikoff, M. (eds.) CAiSE 2003. LNCS, vol. 2681, pp. 587–607. Springer, Heidelberg (2003)
15. Samavi, R., Yu, E., Topaloglou, T.: Strategic reasoning about business models: a conceptual modeling approach. Journal of Information Systems and E-Business Management (2008)
16. Grau, G., Franch, X., Maiden, N.A.M.: A goal based round-trip method for system development. In: Proc. of the 11th Int. Workshop on Requirements Engineering: Foundation For Software Quality, REFSQ, pp. 71–86 (2005)
17. Raadt, V.D., Gordijn, J., Yu, E.: Exploring Web Services from a Business Value Perspective. In: IEEE Joint Int. Conf. on Requirements Engineering, RE 2005, pp. 53–62 (2005)
18. Lespérance, y., Lapouchnian, A.: On Using i* for Modeling Autonomy, Reasoning, and Planning in Adaptive Systems. In: Presentation in istar 2008 Workshop (2008)
19. Oliveira, A.P.A., Prado Leite, J.S.C., Cysneiros, L.M.: AGFL - Agent Goals from Lexicon Eliciting Multi-Agent Systems Intentionality. In: star 2008 Workshop (2008)
20. Kolp, M., Faulkner, S.: The SKwyRL Approach: Social/Spiral Design based on i* and Tropos. In: istar 2005 Workshop (2005)
21. Maiden, N.A.M., Kamdar, N., Bush, D.: Analyzing I* System Models for Dependability Properties: The Uberlingen Accident. In: Proc. of the 12th Int. Workshop on Requirements Engineering: Foundation For Software Quality, REFSQ (2006)

22. Elahi, G., Yu, E.: A Goal Oriented Approach for Modeling and Analyzing Security Trade-Offs. In: Parent, C., Schewe, K.-D., Storey, V.C., Thalheim, B. (eds.) ER 2007. LNCS, vol. 4801, pp. 375–390. Springer, Heidelberg (2007)
23. Strohmaier, M., Yu, E., Horkoff, J., Aranda, J., Easterbrook, S.M.: Analyzing Knowledge Transfer Effectiveness -An Agent-Oriented Modeling Approach. In: Proc. of 40th Hawaii International Conference on Systems Science, HICSS 2007, p. 188 (2007)
24. Yu, E., Strohmaier, M., Deng, X.: Exploring Intentional Modeling and Analysis for Enterprise Architecture. In: Proc. of the EDOC 2006 Conf. Workshop on Trends in Enterprise Architecture Research, TEAR 2006 (2006)
25. Mazón, J., Trujillo, J., Serrano, M., Piattini, M.: Designing data warehouses: From business requirement analysis to multidimensional modeling. In: Proc. of Int. Workshop on Requirements Engineering for Business Needs and IT Alignment (2005)
26. Arzdorf, T., Gans, G., Jarke, M., Lakemeyer, G., Schmitz, D.: SNet: A Modeling and Simulation Environment for Inter-Organizational Networks. In: istar 2005 Workshop (2005)
27. Kolp, M., Do, T.T., Faulkner, S.: Multi-Agent Architecture for E-Business Systems: An Organizational Perspective. In: Proc. of 23rd Int. Conf. on Information Systems (2002)

# Modeling Strategic Alignment Using INSTAL

Laure-Hélène Thevenet[1,2]

[1] CRI, Université Paris 1 Panthéon-Sorbonne 90 rue de Tolbiac 75013 Paris
[2] BNP Paribas, BDDF Informatique, 41 rue de Valmy 93100 Montreuil
laure-helene.thevenet@malix.univ-paris1.fr

**Abstract.** Strategic alignment has emerged as a usual term over the last decades and is now widely used. Yet the definitions available are often vague and lack of formalism to support a systematic reasoning. One difficulty is to deal with high level organization objectives, which are complex by nature, rather informal and cross-cutting. This article presents parts of an Information System (IS) engineering method, called INSTAL, that guides the process of eliciting IS evolution requirements by analysing strategic alignment models. This paper focuses on the upper part of this process that consists in constructing these models. The INSTAL approach is original since it relies on a meta-model that defines alignment formally and it guides the construction of strategic alignment models with complementary techniques. The paper presents these features and their application onto a real case drawn of a French banking company.

**Keywords:** alignment model, information systems, strategic alignment, strategy.

## 1 Introduction

The alignment between Information Systems (IS) and organizations' strategy remains one of the main areas of concern for business executives. Studies show that it was ranked for several years as a top issue faced by organizations. The reason for that is that business executives and CIOs strongly believe that it has an influence on organizational outcomes, performance and competitive advantage.

It is commonly admitted that strategic alignment (SA) is impacted by the changing environment. However, it is still unclear how to achieve and sustain SA over time. Recent researches show that IS alignment -in general- should be considered as a continuous process of co-evolution (e.g. [1], [2], [3], [4]). We believe that this peculiar dimension of alignment, combined with the need for a common understanding to communicate between the strategic and IS worlds, call for novel approaches to cope with SA, in particular to specify it.

Several approaches were already proposed to deal with SA. In requirements engineering it was proposed to represent in a same model the organization's strategic goals and the activity and the processes through which these goals are realized. For Example, the B-SCP approach [5] shows how to deploy strategic goals onto IS requirements. However, SA is not specified or modeled as an individual concept. Similarly, Enterprise Architecture (EA) approaches seldom specify clearly SA (more details can be found in [6]). For instance, in Zachman's EA framework dependencies

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 261–271, 2008.

are drawn between different facets of an organization. In the TOGAF, the organization's strategy is considered as an input: it is identified at early phases but not modeled as the other concepts used are. None of these approaches really propose a formal approach to SA modeling that would support a systematic reasoning about the synergy between organizations' strategies and the IS.

This article focuses on the process of identifying SA links and on their modeling. This part of the INSTAL (INtentional STrategic ALignment) method was designed to face several challenges: (i) the complexity of SA, (ii) the combinatorial explosion of links, (iii) the need for a consistency with other items such as metrics, existing documents, or other views on alignment.

The issues raised for alignment remain valid for SA. However, the singularity of the SA reveals a number of additional issues: (i) conceptual mismatch, (ii) gap in levels of abstraction, (iii) lack of formal conceptual definition of strategy, (iv) antagonism time of SA (strategy being defined in the long-term), (v) combinatorial explosion (strategy having a crosscutting nature), (vi) crossing nature, and (vii) lack of specific indicators. These issues have been considered in developing the method.

The INSTAL method proposes (i) a process model to specify SA, and (ii) a meta-model of SA. The originality of INSTAL stands in its guidelines to explore SA under different perspectives as defined by the meta-model. Besides, the INSTAL meta-model is also novel in that it considers that SA links are complex (not just one-to-one) with several sources and several targets.

The article is organized as follows. Section 2 presents an overview of INSTAL. Section 3 focuses on the alignment meta-model and its four perspectives on SA. Section 4 focuses on the upstream part of the INSTAL process, which guides SA modelling. Section 5 illustrates the approach using an industrial application example developed at BNP Paribas (BNPP). Section 6 presents a discussion and perspectives.

## 2 Overview of the INSTAL Method

The Fig. 1 gives an overview of INSTAL and its four parts described below:

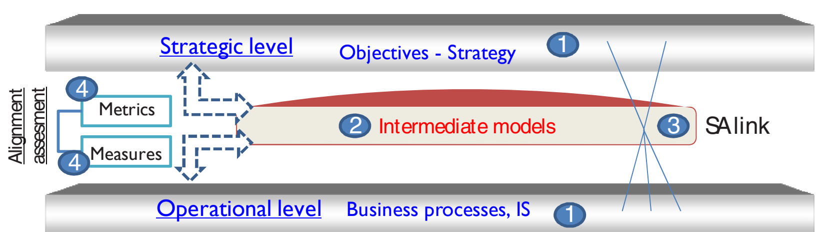



**Fig. 1.** INSTAL method overview

The **organization's part** of the INSTAL method uses documents about strategy definition such as business plans, strategic reports, performance indicators, and operational elements such as IT components, IT functionalities, and business process specifications. It is thus split into two levels: the strategic and operational levels.

The **intermediate model part** of INSTAL provides a unified view on the organization's part. Experience shows that organization strategies are defined in terms of goals, actors and performance indicators and systems are specified using concepts such as objects, events and functions. This conceptual mismatch results in a difficulty to draw links between them. Strategy is also often a cross-cutting cutting concern. As a result, drawing alignment links between the strategic level and the operational level is complex and time taking. More, it can lead to a network of links between multiple elements. INSTAL proposes to use intermediate models to state in an abstract way how the elements from the operational level tend to contribute to the strategy.

INSTAL's **alignment links part** uses alignment links that relate elements from the strategic and operational levels. SA links aim at highlighting how operational elements contribute or not to the alignment link with elements of the strategy. As detailed in the next section, we defined complex links semantically rich: analyzing alignment links allows highlighting cases of alignment, as well as misalignment.

INSTAL proposes in its **metrics/measures part** a quantitative view on SA though valuations of the satisfaction of objectives. Our observations at BNPP showed us cases in which - although all the needed operational elements were well deployed to support a strategic objective - the target objectives were still not attained, for example because other complementary actions had not been taken (e.g. advertising, communication, price revision). Metrics and measures (found in documents) help evaluating SA and complete SA models.

The methodological process of INSTAL is composed of two main goals that are to: *model SA* in order to diagnose current SA and *define evolution requirements*. The method provides guidance to achieve these goals, only the guidance to model SA will be detailed in this article (section 4 and 5). The outputs of the method are evolution requirements to obtain a future situation conform to SA requirements.

## 3 Strategic Alignment Meta-model

INSTAL exploits four perspectives on SA but the INSTAL's formal definition of SA relies first on an alignment meta-model.

### 3.1 Alignment Meta-model

INSTAL is able to define SA under the form of complex links. As the left hand part of Fig. 2 shows it, several kinds of complex links can be considered: rake, brush and spider's web. As shown in the meta-model (right hand part of Fig. 2), our approach to define complex links is to aggregate their source and target ends, rather than to aggregate several simple links. Indeed, we observed in practice that several simple links can play different roles in different aggregations with other links. This approach has the advantage of allowing defining specific roles played by ends in SA links.

A simple link relates one element at the source end and one element at the target end. Complex links establish a link between one or several source elements and one or several target elements. Complex links help managing complex cross cutting concerns, which often impact numerous elements at the operational level. Instead of defining (n x m) simple links between m cross-cutting requirements and n operational

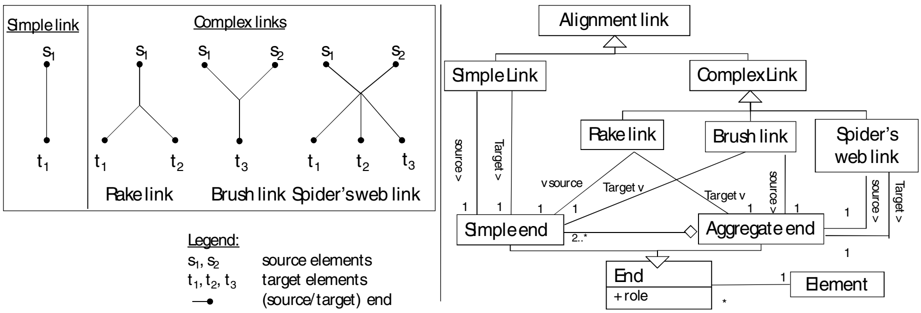



**Fig. 2.** The four types of alignment links (left) and the alignment meta-model (right)

requirements, INSTAL proposes to define a unique link between the aggregated m strategic elements and the aggregated n operational elements. Differentiating complex links from simple links is not enough to formalise SA, as they do not provide any specific semantics. INSTAL's extensions to the alignment meta-model fill this gap.

### 3.2 The Four Extensions in the SA Meta-model

**Organization extension:** elements correspond to the ends in the strategic alignment model (in Fig. 2). The constraint is that strategic elements are involved in source ends and operational elements in target ends of the SA links.

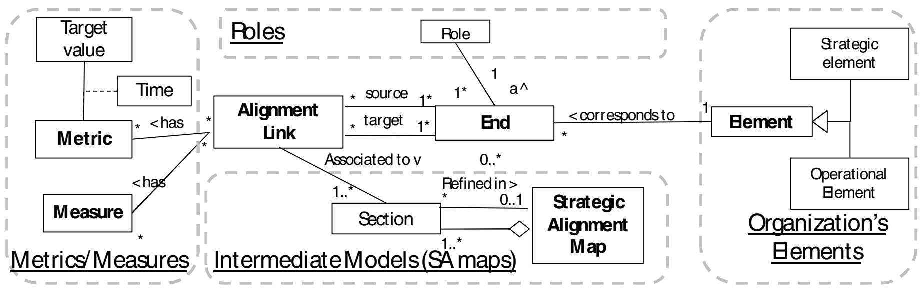



**Fig. 3.** INSTAL's SA meta-model (simplified) with the four extensions

**Role extension:** each end element has a role in a SA link. Five types of roles are proposed. They can be combined to specify the semantics of the SA links through their ends. These roles were defined in INSTAL using an action research approach then tested on several case studies [7][8]: *Necessary*, *Useful*, *Sufficient, Constrained*, *Contradictory* (and combination such as *necessary and sufficient*).

**Intermediate models extension:** as already shown in [7][8], INSTAL provides a way to specify intended SA with intermediate models using the MAP formalism (they are thus maps). This approach was found particularly helpful to manage the different granularity levels, unify multiple points of views (as for business/IS alignment in [9]), and deal with the conceptual mismatch between the strategic and operational levels.

Maps are composed of sections that are triplets themselves composed of two goals (the source and target goals), and a tactic. The source goal of the section should be achieved (i.e. satisfied or satisficed) to be able to undertake the section. The section is achieved when the target goal is achieved. Tactics tell how the target goal can be achieved. Sections can be refined in a map at a lower level; this refinement mechanism allows to describe SA at different granularity levels and to organize goals and tactics hierarchically. Maps in the INSTAL method are named SA maps and the sections provide an ambivalent view on strategic and operational levels (common intentions). Each section can be associated to an SA link.

**Metrics and measures extension:** a SA link can carry one or several metrics and measures. In practice, metrics apply to strategic elements and measures to operational elements. They are attached to links and not to ends because it can be necessary to define them even if the SA link is not complete.

## 4 INSTAL Process Model for Guiding SA Modelling

This section focuses on the identification of SA links. Guidelines are briefly presented and illustrated in the next section. Four complementary tactics are proposed to help in identifying SA links, and presented in Fig. 4 with MAP formalism. This map details how modelling SA, it refines a section of a higher level map (with the goals: model SA and define evolution requirements). Only the tactics to specify SA links are mentioned since we focus on these ones in this article.

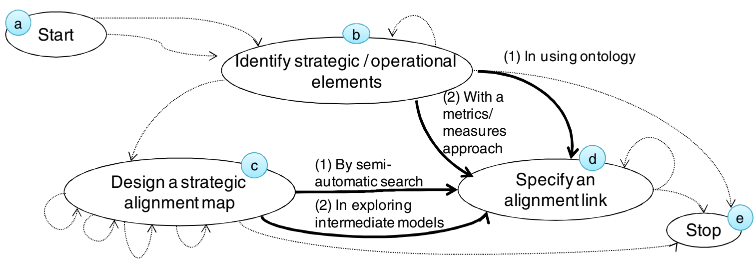



**Fig. 4.** INSTAL process model to model strategic alignment

The first tactic is based on ontology use and requires a good maturity level. The second one is more pragmatic, based on documents search tool. The third and fourth ones use respectively SA maps and metrics/measures.

### 4.1 Ontology-Based Tactic for SA Link Identification

Ontology gives a formal definition of real world phenomena as well as their properties, links, constraints and behaviours. INSTAL uses ontologies to identify elements in the two aligned levels, and help draw links between them. Elements are searched for by their type, and links are searched for between types of elements that usually relate. Ontologies provide thus a way to develop a well-defined and agreed view of the business and systems models.

The principle of the ontology-based tactic to identify SA links is to look for SA links by instantiating the concepts defined in the two proposed ontologies. Once this done, it is possible to look for SA link by considering only those links that are likely to be meaningful according the semantics of the instantiated concept. Rather than defining a new ontology, we can exploit existing ones: the business model ontology defined by Osterwalder [10] for the strategic level and the business meta-model defined in COMET [11] for the operational level. The table below summarizes some generic correspondences between elements of the two chosen ontologies.

It can be used to propose links in a systematic way. Once identified, SA links must be evaluated according to their representativeness. The semantics of SA links can then be specified by defining the roles of the linked items.

**Table 1.** Example of correspondences between strategic and operational level elements

| Elements at the strategic level | Elements at the operational level |
|---|---|
| Value Proposition | corresponds to **business processes**. Global business processes can describe general value propositions, detailed business processes describe offerings. |
| Target customer | corresponds to an **external actor resource**. Different customer segments exist and are characterized by different properties |
| Capabilities | corresponds to **recurrent business processes** (similar business processes that have few variants) **or steps** depending on the granularity of this one. |

## 4.2 Textual Search Tool Tactic for SA Link Identification

When a SA map has been specified to provide an intentional view on SA, the textual search tool tactic can be used to identify new SA links. INSTAL recommends to start the search using the terms of the sections' label, i.e. its source goal (except with start), its target goal (except if stop), and its tactic. The basic algorithm is the following:

For each term:
1. delete conjunctions, determiners and reserved keywords of the search tool
2. add synonyms, hyperonyms and hyponyms --this enriches the research criteria
3. search in strategic and operational documents

Some search tools can be used to achieve the search. A SA link is identified when relevant references are found both in documents from the strategic level and documents from the operational level. A second algorithm can also be used:

For each term:
1'. delete conjunctions, determiners and reserved keywords of the search tool
2'. use antonym or express opposite terms
3'. search in strategic and operational documents

For example, 'develop multi producer application' can be replaced by 'develop mono producer application' or 'develop application for one specific producer'.

### 4.3 Use of Map Models for SA Link Identification

The analysis of SA maps helps identifying new SA links. Maps are an intentional entry point to SA links. Moreover, a section of a SA map can be refined in another map at a lower level. The analysis of SA maps through refinement allows a black-box/white-box approach for the analysis and discovery of SA links.

For example when a spider web SA link relates to a section of a SA map which is refined into a more detailed map, the SA link can itself be decomposed. This is achieved by replacing ends to coarse grain elements by aggregate ends with finer-grain elements that decompose them, as indicated by the SA maps.

Another example is when the SA map shows complementary sections that require each other or alternative sections that are mutually exclusive. If one of them relates to a SA link then the link should fulfill the constraints imposed by the dependencies.

### 4.4 Metrics and Measures for SA Link Identification

Metrics explain how to evaluate the achievement of strategies. Measures indicates how monitoring is achieved at the operational level. When a metric is associated to a SA link, then the operational elements needed to evaluate the metric should be targeted by the link. The other way round, the collection of measures associated with a SA link should be fully covered by the strategic elements that are at its source. If a measure appears in the link and does not seem to monitor any of the source strategic elements, then either it is not needed, or there is a hidden or implicit strategic element. In the latter case, a strategic element should be added to the SA link.

## 5 Real Case Example

The INSTAL method was applied on the case of SA at the French Retail Banking (FRB) of BNPP. The FRB proposes products and services throughout France adapted to the different customers segments. Since few years, the FRB positions as a distributor of financial solutions. It has developed partnerships, contrary to the past when it ensured both distribution and production. The development of partnerships allows distributing a wide range of financial and associated products such as insurances (e.g. savings, protection, and housing assurance) and credits.

### 5.1 The FRB BNP Paribas SA Map

Fig. 5 presents a SA map that helps to have a common understanding on SA.

Different documents presenting the organization were considered and a collection of operational documents. The study of the FRB values and aspirations made emerge three main goals present in this map: (b) *Attract new clients and subscriptions*, (c) *Optimize the offers (products and services*) and (d) *Improve the clients' satisfaction*.

The FRB wants to optimize the offers (ac1) by developing the home banking and multi channels (e.g. ATM, branches, internet), (ac2) by developing the cross selling, (ac3) by developing product innovation, (ac4) by partnerships development etc.

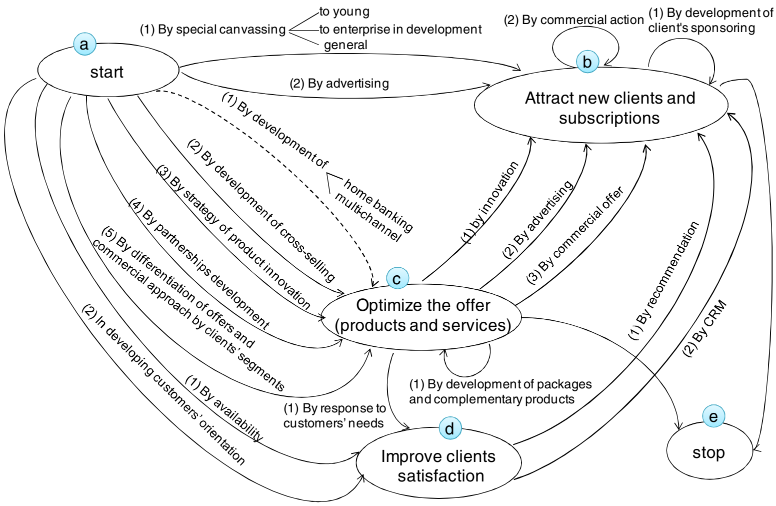



**Fig. 5.** An example of SA map representing the intended SA at FRB BNP Paribas

An important goal is also the improvement of customers' satisfaction since a satisfied customer may speak in good terms of its bank and so naturally recommend it. Satisfied customers will stay loyal and it will be easier to sell them products and services. The last important goal is to attract new customers and subscriptions.

## 5.2 Identifying SA Links Using Ontologies

In Table 2 the two levels to align are organized according to the ontologies, which makes easier to identify links between them.

**Table 2.** Example of SA links identification using ontologies

| Ontology elements at the strategic level | Ontology elements at the operational level |
|---|---|
| **Value propositions:**<br>- Offers products and services to customers, categories of products and services are: saving and deposit, financial and market investments, means of payment, asset management…<br>- Develop partnerships to externalize production in order to diversify offers (e.g. partnerships with insurance companies) | -Links with business process, some links can be directly discovered with general abstract business process such as « offer financing products and services » and the refined business processes by products type (e.g. revolving credits, redeemable credits), clients segment (e.g. distribute consumer credits to individuals clients), distribution channel etc. Typologies of products and services can be useful to map the value proposition to the concerned business processes.<br>- activities in the business process that correspond to production activities and that are assumed by external actors can be mapped with the value proposition consisting in developing partnerships. |
| **Target customer:**<br>Individual clients, entrepreneurs and freelance processionals, and corporate clients. Target customers can also be defined in terms of relationship, for example the customers' children. | Individual clients, entrepreneurs and freelance processionals, and corporate clients.<br>At the operational level distinction is made between physical person and artificial person (e.g. corporation). Attributes allow defining specific customers categories. |
| **Capabilities**<br>e.g. products management, customer management, sales, operations management, marketing… | The FRB capabilities are diverse, business processes or steps correspond and detail how the FRB uses its capabilities, and the capabilities and resources of partners, e.g. for the distribution of consumer credits the FRB uses the expert system of a credit partner to assess credit risk and decide on the credit. |

As shown in Table 2, it is possible to find steps of business process corresponding to production activities and verify if partners are responsible of some activities. In this case, multiple links are specified between elements but the purpose of the links is not explicitly documented. It is one of the reasons we associated them to intermediate models. In the case of the partnerships development, ac4 in Fig. 5 will be used as an anchor to the SA link involving the concerned strategic and operational elements.

### 5.3 Identifying SA Links Using Search Engine

The "Copernic Desktop Search" tool was used to perform the search and "WordNet Search 3.0" to find related word such as antonyms and synonyms. Otherwise, the tactic was implemented manually.

Applying the search tool tactic to the section ac1 generated a SA link. To achieve this, the search was applied to the selected strategic and operational documents using the query "Optimize offers products services developing home banking multi channels branches website ATM Minitel". Some terms were added because they correspond to various kinds of channels in the multi-channel strategy. Several references were found in the strategic documents. For example, the search result pointed out to a document that contained the texts: "... increase in client contact through the Multi-channel Retail Banking system, which includes branches, the telephone (via the Client Relations Centre), and the internet, through bnpparibas.net...", "....number of contact opportunities offered to the various channels...", "...home banking services..." etc. Other references were found by the search tool in the operational documents, they contained the following texts "...number of operations by channels...", "...ATM…", "...internet...", "...branches...", which seem to fit with the findings at the strategic level.

Table 3 summarizes some ends (for SA links) identified using the search tool tactics. The '-' indicates negative links in terms of contribution (i.e. contradictory) and the '(*)' indicates the text lacks of precision. A lower granularity level analysis is necessary to define the SA links precisely and decide if they are representative enough of the SA. A first set of links can be proposed but it has to be checked in order to eliminate bad proposals. For example, the development of partnerships (ac4) can be applied only in some business cores, the negative roles are thus to be checked.

**Table 3.** Results obtained

| Sections | Goal to attain | Number of contribution links towards | |
|---|---|---|---|
| | | Strategic documents | Operational documents |
| ac1 | C "optimize the *offers(products and services*" | 4 | 10 |
| ac2 | | 3 | 3 (*) |
| ac3 | | 2 | 4 (*) |
| ac4 | | 3 | 5 (*) / -3 (*) |
| ac5 | | 5 | 12 |

### 5.4 Consistency of the SA Links with BNPP Metrics

Fig. 6 illustrates the result of the SA link modelling activity with an extract that focuses on a complex SA link between two strategic elements and three operational elements. Roles are defined at each end of the link to specify the role played by each element in the SA. Besides, the SA link is associated to section ac1 of the FRB SA Map, and to metrics and measures that helped specify it in a more complete way.

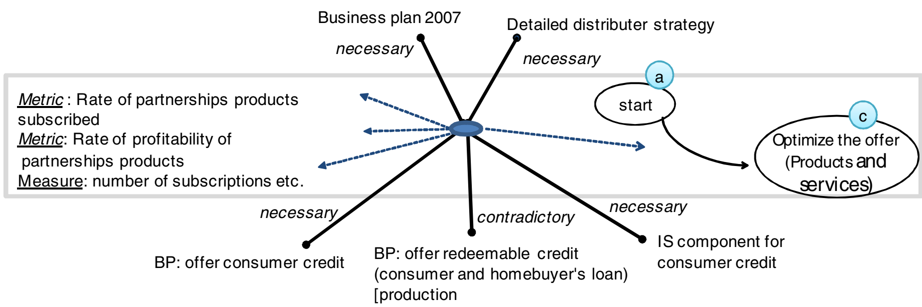



**Fig. 6.** SA link example

The comparison between the target value of the metric and the sum of the effective measures allows knowing if the quantitative objective is (or can be) attained.

The analysis of the consistency between operationalisation of metrics and operational elements, on which the corresponding measures should be monitored, shall allow identifying supplementary SA links.

# 6 Conclusion

This paper has (i) presented the alignment meta-model and its extensions for the SA modeling with the INSTAL method, (ii) shown how to specify complex links to document organization's strategic and operational elements involvement in SA, and (iii) described a methodological process that guides the identification and specification of SA models and SA links.

The paper reports an example of application of INSTAL onto a real life case study. Documents were used to identify SA links and model them using three of the four guidelines proposed. Complementary results are:

- Using ontologies required additional work that could have been reduced if the organization had had some maturity with ontologies. The ontologies allowed organizing strategic and operational elements, and verifying to some extent that no important elements were forgotten in the intermediate models.
- Modeling SA map helps localising roughly the SA links. They provide a good starting point to start automated searches. Supplementary techniques are however needed to make the search more precise and efficient.
- The use of metrics and measures was a good supplement to the other tactics. It confirmed that even when all the needed operational elements seem to be well deployed, it still happens that the target quantifiable objectives (defined by metrics) are not attained. This revealed the need to reinforce the operational elements involved or add new evolution requirements.

We are exploring the complementarity of SA models and metrics for the analysis of SA, as initiated in [12]. Besides, our research activity focuses in parallel on the elicitation of IS evolution requirements based on systematic analyzes of SA links.

## References


1. Sabherwal, R., Hirschheim, R., Goles, T.: The Dynamics of Alignment: Insights from a Punctuated Equilibrium Model, Organization Science, vol. 12(2) (March-April 2001)
2. Chan, Y.: Why haven't we mastered alignment? The Importance of the Informal Organization Structure. MIS Quarterly Executive 1(2), 97–112 (2002)
3. Etien, A., Salinesi, C.: Managing Requirements in a Co-evolution Context. In: International Conference on Requirement Engineering (RE), Paris, France, pp. 125–134 (2005)
4. Etien, A.: Business Process / Information System Co-evolution. ISI journal, Hermés (2008)
5. Bleistein, S., Cox, K., Verner, J., Phalp, K.: B-SCP: a requirements analysis framework for validating Strategic Alignment. Information and Software Technology 48(9) (2006)
6. Salinesi, C.: Enterprise Architecture: from Practice Issues to Research Innovation, Research Challenges in Information Science (RCIS), Ouarzazate, Morrocco, April 2007, pp. 107–120 (2007)
7. Thevenet, L.H., Salinesi, C.: Aligning IS to organization's strategy: the INSTAL method. In: Krogstie, J., Opdahl, A., Sindre, G. (eds.) CAiSE 2007. LNCS, vol. 4495, pp. 203–217. Springer, Heidelberg (2007)
8. Thevenet, L.H., Rolland, C., Salinesi, C.: Alignement de la stratégie et de l'organisation: Présentation de la méthode INSTAL. Revue Ingénierie des Systèmes d'Information Special Issue on Information System Evolution, Hermès (2008)
9. Salinesi, C., Rolland, C.: Fitting Business Models to Software Functionality: Exploring the Fitness Relationship. In: Eder, J., Missikoff, M. (eds.) CAiSE 2003. LNCS, vol. 2681, p. 2003. Springer, Heidelberg (2003)
10. Osterwalder, A.: The Business Model Ontology a proposition in a design science approach, Ph.D thesis, HEC de l'Université de Lausanne (2004)
11. COMET, http://www.modelbased.net/comet
12. Viscusi, G., Thevenet, L.H., Salinesi, C.: Strategic alignment in the Context of e-Services an Empirical Investigation of INSTAL with the Italian e-Governement Initiative Case Study. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, Springer, Heidelberg (2008)

# Requirements Engineering for Distributed Development Using Software Agents

Miriam Sayão[1], Aluízio Haendchen Filho[2], and Hércules Antonio do Prado[3,4]

[1] PUC-RS - Pontifical Catholic University at Rio Grande do Sul, RS, Brazil
[2] Anglo-Americano - Anglo-Americano College, Rio de Janeiro, RJ, Brazil
[3] Embrapa Food Technology, Rio de Janeiro, RJ, Brazil
[4] Catholic University of Brasília, Brasília, DF, Brazil
miriam.sayao@pucrs.br, aluizioh@terra.com.br, hercules@ctaa.embrapa.br, hercules@ucb.br

**Abstract.** The global spread of business is introducing a new trend in the organization of work. This dynamics in business has led to a distribution of activities in different locations, affecting also the way people develop software. Developers, software engineers, quality inspectors, and also users and customers, are distributed around the world. Since stakeholders are distributed, the software development cannot be efficiently performed without a support for articulating these people in a consistent way. Issues like distance, communication, and different time zones introduce additional difficulties to the stakeholders involved in the process. This paper explores the use of an agents architecture designed to support the Requirements Engineering, specifically the Verification and Validation (V&V) activities, in the distributed development process. A goal-driven approach is used to define high-level V&V goals that, after refined, makes possible to derive requirements and assigning responsibilities to the actors (humans, software agents, devices and programs).

**Keywords:** Requirements Engineering, Requirements Verification, Distributed Software Development, Software Agents, Goals Models.

## 1 Introduction

The complexity inherent to the software development process increases when stakeholders are geographically distributed. It happens due to factors such as distance, different language skills, and delays in communication. This process requires the interaction of different players, like users, customers, developers, requirements engineers, project coordinator, and quality inspectors. Each of these actors plays a role in the process, aiming to achieve their own goals and all of them aiming to reach a common goal. The common goal is to obtain, at the end of the development process, a product that meets the needs of customers and users, with the quality desired.

Problems like this have been successfully approached by applying the agent-oriented paradigm. Complex systems in which stakeholders perform a variety of roles, interacting to achieve a common goal are natural candidates for agent-based solutions. Examples of agents' application in software engineering can be found in the software development process [5, 10, 11] and in the requirements engineering [15].

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 272–281, 2008.

We therefore propose the use of agents' paradigm to support activities related to requirements process, specifically addressing the requirements verification and validation (V&V). We argue that the agents' paradigm is suitable to approach problems in the distributed development software process. In our view, difficulties regarding to disparate location of stakeholders, communication among them, and differences in time zones, can find a natural solution when adopting agents based solutions. It includes issues like traceability and management of requirements evolution along the distributed development process, sharing effective information related to the system under development.

The distributed requirements engineering process needs supports from new techniques and tools in order to make easier the requirements development and management. This paper explores the use of software agents structured and designed to help the Requirements Engineering (RE) in distributed development. A goal-driven approach is used to define high-level V&V goals that, after refined, makes possible to derive requirements and responsibilities assignments to the actors (humans, software agents, devices and programs).

In the next sections we present a background on the main technologies involved, the proposed approach describing a goal-task decomposition, some design details of the architecture implementation, and a discussion about the contribution of the approach to the RE context.

## 2 Background

In the global economy, the software development by geographically distributed teams is becoming a rule. Since the 90`s the distributed software development is increasing in importance for software engineering researchers. The additional problems caused by stakeholders geographically dispersed are reported in many research papers and books [2, 17]. In this scenario, requirements verification and validation for medium or high complexity software can involve the treatment of hundreds or even thousand requirements. With this complexity order it is important to provide computational support for the software engineer execute quality activities. We propose a strategy which combines natural language processing (NLP) techniques and software agents to support verification and validation activities; in this work we focus on the use of software agents to support V&V activities.

The agent-oriented paradigm for software development has been applied in distributed artificial intelligence since the early 1980s [8]. Despite the existence of conflicting definitions for agents, according to Wooldridge [18], for many authors their autonomy characteristic is regarded as essential. The term autonomy is used to indicate that agents run not necessarily requiring the intervention of human users or other systems. Applications using agents are found in different areas, and more recently in the Software Engineering field [6] [11] [13].

Wooldridge [18] defines software agent as a system for computer placed in any environment and able to take autonomous actions on the environment in order to achieve the desired goals. The agent monitors the environment by means of sensors, and can trigger actions that will occur on the same environment. Another definition of agents found in the literature, and that will be used in the context of this work, establishes that agent is an interactive, adaptive and autonomous entity that has a "mental state". Jennings et al. [14] argue that the agent-oriented technology is still in an early stage, but will have a significant future in software engineering. We believe that an agent based approach can be useful in the context of distributed requirements process due to the following characteristics:

a) Autonomy: agents can monitor the environment, detecting relevant changes or events and reporting them to the involved people. In this work, agents can monitor changes in requirements artifacts, communicating automatically these changes to stakeholders.
b) Flexibility: agents can be dynamically inserted/excluded transparently in/from the environment of implementation. The set of stakeholders evolve throughout the requirements process, and the set of agents may reflect this trend.
c) Interactivity: agents can communicate by means of messages; communication activities among stakeholders are essential to the activities of requirements management and agents may play part of these roles.
d) Integration of distributed knowledge: agents can collaborate in activities that require integration of knowledge for decision-making. The knowledge related to the system is geographically distributed among the dispersed stakeholders, and agents can collaborate in that integration.
e) Rationality: knowledge of stakeholders in the requirements process can be easily represented in the agents' model.

In our approach we explore agents' autonomy and flexibility, and also communications among both software agents and software agents and humans. We apply goal-oriented techniques in order to identify objectives for software agents.

Goal-oriented techniques are used for requirements analysis and had provided a consistent way to refine organizational and technical objectives, which make it possible to achieve these objectives more effectively. The explicit modeling of goals and the integration of goals in the multi-agent systems (MAS) development have been widely considered in the literature. According to Evans [7], a goal is a set of states of the world that an agent is committed to achieve/maintain. Therefore a goal is a situation, but not all situations are goals. A set of states of the world is not always a goal unless an agent is committed to achieve/maintain this set of states. Requirements engineering is concerned with the identification of high-level goals to be achieved by the system envisioned, the refinement of such goals, and the operationalization of goals into services and constraints [4]. For this purpose, responsibilities from the resulting requirements are assigned to actors, such as humans, software agents, devices and programs.

There are some approaches that can be used to define and refine goals. The i* notation [20] has been used for modeling systems, aiming to show the intent of the actors, the dependencies between them and the non-functional requirements (NFR). For example, KAOS methodology [4] is aimed at supporting the whole process of requirements elaboration - from the high-level goals that should be achieved by the composite system to the operations, objects and constraints to be implemented. After selecting a set of alternatives to achieve these objectives, one can elaborate on them during subsequent phases to make them more precise and complete. The goal refinement structure for a given system can be represented by an AND/OR directed acyclic graph. For this work, we use the i* notation to represent the more abstract views of goals, actors, NFRs, and dependencies and the KAOS notation to represent the lowest level decomposition. KAOS and i* approaches were both applied due to the fact that, in our point of view, the former is more suitable to represent low level details and the latter presents the ability to represent the *softgoals* and dependencies among them and actors.

# 3 Proposed Approach

The main focus of our approach is the Verification and Validation (V&V) phases, considering the full process of RE. During the V&V different stakeholders such as users, customers, developers, requirements engineers, project coordinator, and quality inspectors interact aiming to verify requirements quality and to assure they meet customers and users needs. Each of these actors plays a role in the process, aiming to achieve their own goals and a common goal: to obtain at the end of the development process a quality product that satisfies customers and users. We are now interested in focusing the software agents architecture and the communication among stakeholders (human agents). Software agents can be seen as personal assistants of different stakeholders in the V&V process, providing effective distribution of work in the RE process context, reducing the communication problems among real actors. In order to illustrate our approach, Figure 1 presents the model of strategic dependency i* [20] in the processes of requirements verification and validation.

This model illustrates the goals of each agent and the dependencies among agents in the achievement of their goals. The main actors (customer, developer, engineer requirements, project manager, and inspector) have relations of dependency for resources, for *hard goals* (goals) or *softgoals*, which represent the intention of the actors. Unlike the (hard) goals satisfaction, which can be established by means of verification techniques, *softgoals* satisfaction cannot be established in a clear-cut sense (used to model non-functional characteristics of the system). The dependencies between goals and *softgoals* can be established by using the NFR framework [3]. Once specified the model of strategic dependency, the next step is the goals refinement. The focus of this paper is the (hard) goals. A goal is decomposed in sub-goals and Commitments, and commitments are assigned to the responsibility of an agent. The notation employed here comes from KAOS methodology [4]. The example in Figure 2 shows the refinement of the SRS Verified goal into sub-goals and tasks.

The SRS Verified goal is decomposed into two sub-goals, Inspector Verify SRS and Req. Eng Verify SRS, such that the satisfaction of these sub-goals entails the satisfaction of the parent goal. These sub-goals are decomposed into tasks. Goal decomposition should stop when a task performed by an agent meets the satisfaction condition. The next step in the Goal-Task decomposition is to define the tasks that must be performed in order to satisfy the sub-goals. We define task according to Evans [7] as an activity that can be decomposed. First, high-level tasks are defined and afterwards they must be decomposed into sub-tasks. Notice that a goal describes motivation, while a task describes activity. Prepare Verification task is decomposed into two sub-tasks: Receive Notify and Initial Report. These sub-tasks involve the

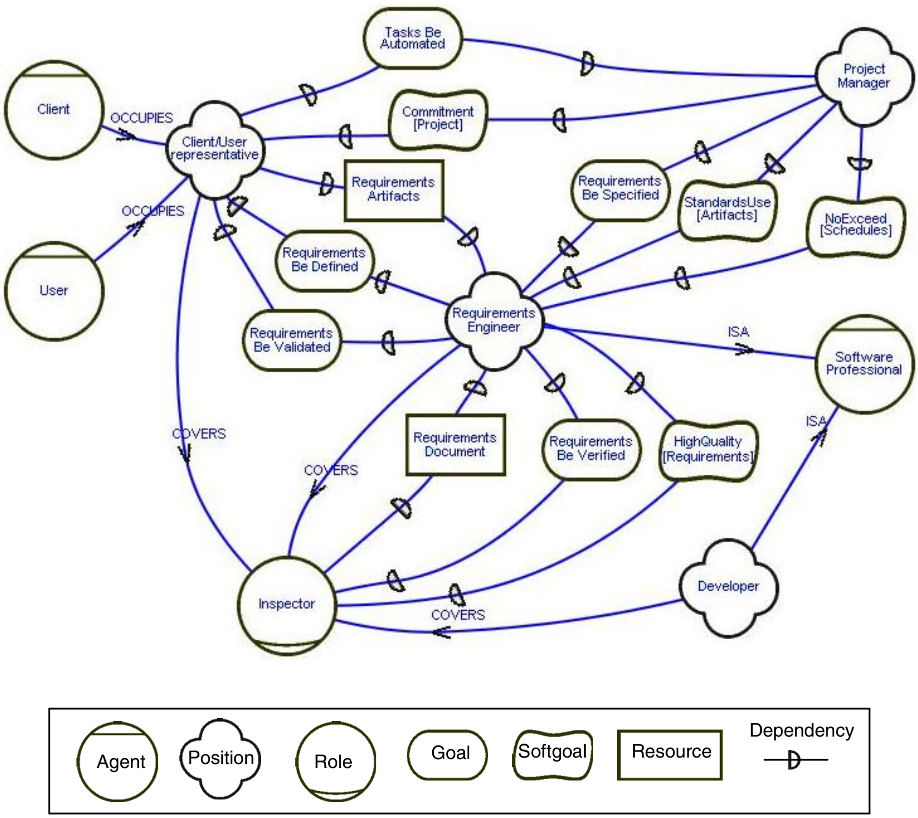



**Fig. 1.** Strategic dependence model for verification and validation of requirements

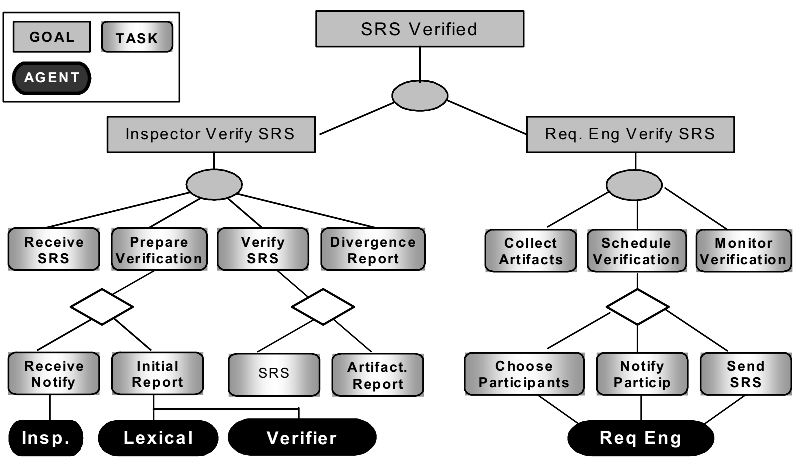



**Fig. 2.** Goal-Task decomposition for the SRS Verified goal

direct commitment of four agents: Inspector (Insp), Lexical, Verifier and Requirements Engineer (Req Eng). The Inspector and Req Eng agent represents one or more instances of the actors involved.

During the requirements verification process, developers, tests engineer, customers, and requirement engineer evaluate the SRS and interact aiming at ensuring that the requirements in the SRS match standards and rules and are correctly modeled. Each inspector brings the vision of the group to which s/he belongs: software developers, software testers, architectural designers, and clients. In this process we visualize agents for identifying whether the intentions of the actor he represents are reflected in the document being evaluated. On the next level, the tasks are decomposed into actions. Due to space limitations, we go deeper only in the Initial Report task and in the Lexical and Verifier agents rather than others, since both are more related to the application and lead us to a more focused discussion. Next, the actions to be performed by the agent are defined. The task Initial Report is sequentially decomposed into six actions (Figure 3). Action is an atomic unit of a function that modifies the status of the system and the environment. The agent committed to a task or sub-task is responsible for the implementation of all actions decomposed from this task. For this purpose, it can request collaboration from other agents to perform actions.

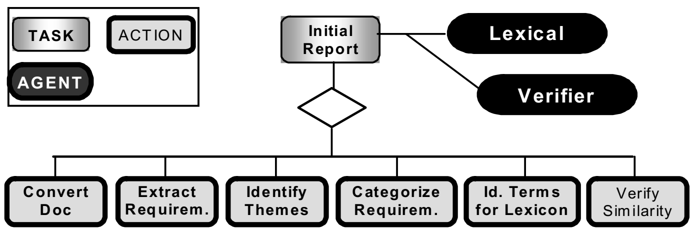



**Fig. 3.** Decomposition of the Initial Report task

The Lexical agent plays the roles related to the text handling tasks like: (a) relevant theme identification, (b) identification of requirements or other models (such as use cases or test cases) associated with a particular theme, and (c) identification of issues relevant to textual artifacts (requirements documents, electronic messages, vision documents, the test cases). Beyond performing these tasks, the Lexical agent also creates the traceability matrix RNF x RF. To accomplish its objectives, the Lexical agent applies Natural Language Processing techniques. The Lexical agent collaborates with the Lexical Constructor, Tracker, Visions Generator, Observer, Validator, and Verifier. In our example, the Lexical agent centralizes the services of lexicon treatment that is performed upon the requirements document. It is committed to examine different types of documents, extracting, sorting, and organizing information to generate an initial requirements document and an artifacts report. It provides services that can be used for several agents in the system, taking requests for documents conversion, extraction of words' stems, get contexts, identifying collocations, categorizing requirements based on relevant themes [1], and identifying terms not found in the lexical dictionary.

The Verifier agent is responsible for checking the requirements activities, like identifying requirements in duplicity and omissions in NFRs. In the verification process, a possible activity is to inspect the requirements document. The process of inspection is characterized by the use of a reading technique applied to an artifact, aiming to locate discrepancies, errors or omissions, according to pre-established criteria. For identifying duplicate requirements, a common error that occur when the stakeholders are distributed, requirements are extracted from SRS, generating the term-document matrix and calculating the similarity rates of Dice, Jaccard and cosin [16]. For omissions in NFRs a content based analysis technique is used, that is not presented here since the technique is not yet implemented as an agent service. In order to play these responsibilities, the Verifier agent collaborates with the Lexical, Statistics and Communicator agents. Other functions are going to be included in this agent in order to extend the offered services. For short, the Verifier agent is responsible for checking the overall requirements quality.

After identifying the agents and actions they must perform, the MAS design can be developed. The roles model [19] can properly be used to represent the rights and responsibilities of agents as well as the actions they must perform and its collaborations. The agents' lifecycle, its internal behavior and interaction can be modeled by using the conventional UML (Unified Modeling Language) tools. The lifecycle phases of the agents (or applications) can be represented using the activity diagram. The agent internal behavior can be designed using the state diagram and the interactions among agents can be modeled by using interaction diagrams.

## 4 Architecture Implementation Details

The agents were instantiating by using MIDAS [12], a MAS development framework specified under the service-oriented architecture (SOA). The platform provides a complete execution environment in where agents can execute concurrently in the same host and communicate with external applications. The communication among the client machines and local or front-end servers is performed via HTTP protocol, enabling any machine connected to the Internet access the information stored in the container or central server using a browser. External heterogeneous systems can communicate in a bi-directional way with the MAS via Web Services using the SOAP protocol. The mechanisms embedded in MIDAS work into two abstraction levels: (i) in the generic architectural level, providing infrastructure services, such as message transport (send/receive, pack/unpack, requests translate), the control over the platform and agents' lifecycle, tasks for services handling (services registry, publication and discovery), and (ii) in the agents' design level, providing abstract classes that define the hot-spots from which specific features of the concrete agents (or components) can be implemented and a blackboard. The Blackboard [9] is a software entity widely used in symbolic cognitive multi-agent systems. It keeps a set of global variables that controls the environment state and the state of the problem solving.

The platform provides GUI wizards that allow the registering and controlling the agents and managing the MAS lifecycle. Figure 4 shows a GUI of the MAS with the agents and tasks modeled in the previous section. The procedures to synchronize container and agents' lifecycle can be hold at the upper horizontal bar by using the

Main and Agents options. On the right side, the Structure frame shows the hierarchy of resources, identifying agents, components and services taking part of the application. The hierarchy is navigable, and for each selected entity, specifications details are shown in the Details frame placed on the left side. The example shows the details of services provided by the Lexical agent. The actions performed by the agents are represented as services.

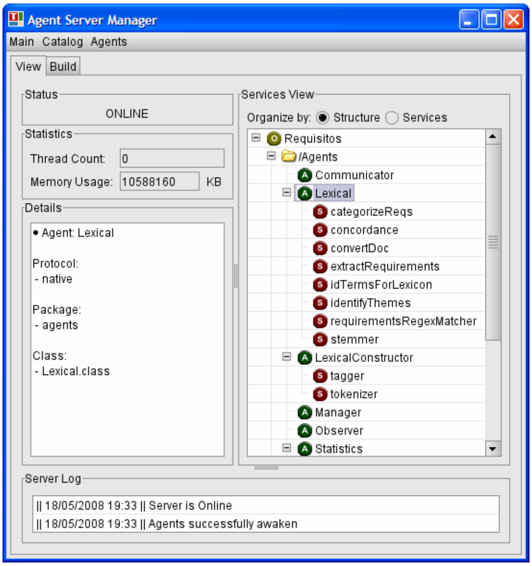



**Fig. 4.** View of the Multi-Agent System in the platform GUI wizard

Agents use the blackboard to capture the environment state, and the implementation kernel is characterized by loops that consult the status of the verificationState variable in a blackboard. The actions sequence of the Lexical agent begins when the Requirements Engineer feels that it is time to start the verification process. After that, s/he defines a date and selects the participants. So, a personal agent representing the requirements engineer sends a message to the participants, in collaboration with the Communicator agent. The Verifier agent is also communicated and performs an initial pre-verification task. This initial pre-verification includes the identification of requirements with high rates of similarity, that we call "duplicate requirements candidates". After that, the Verifier agent sends this initial report to inspectors that analyses and use it as a basis for their reports.

The Communicator agent is responsible for the transmission of messages to the stakeholders, notifying them with respect to events of interest, such as definition of calendars for the processes of verification/validation or changes in the system requirements. The Lexical Constructor agent is responsible for maintaining the lexicon of the application, seeking to update the knowledge base for the organization domain. The Manager agent is a personal assistant of the project manager, being responsible for monitoring the activities of the project manager, assisting it in the scheduling of the V&V activities. The Observer agent is responsible for monitoring the transactions described in the requirements artifacts and, in case of change, is responsible for notifying those involved via Communicator help. Other agents involved in the architecture but not directly related to the scope of this work were not discussed in the text. Further details can be found in [17].

## 5 Conclusion

The research field of Distributed Software Development has received more attention from the academy in the last ten years, with an expressive amount of publications, many of which involving case studies, experiments, industrial reports, and surveys. However, after stressing the literature we could only find scarce results related to RE in distributed development which make use of software agents. This work proposes the use of MAS architecture to minimize the difficulties that arise in the distributed requirement engineering process for applications development. The goal-driven approach, used to specify the architecture requirements, has provided a roadmap for the requirements specification. The goals of stakeholders and software agents are well defined as goals models and both the Strategic Dependency model of i* and the KAOS model was applied to our aiming. While i* represent the highest-level views of goals with its constraints and dependencies, KAOS defines the lower-level tasks, actions and objects to be implemented. The decomposition of the goals in sub-goals, tasks and actions has facilitated the definition of roles and assignment of responsibilities to the agents and stakeholders.

Three main contributions can be considered in our approach: (i) the use of software agents to automate some actions necessary to the V&V process, (ii) the use of different techniques for natural language processing in order to evaluate generated or manipulated artifacts in RE, and (iii) the design and implementation of an architecture for the proposed approach. Our prototype was developed to demonstrate that V&V tasks can be partially automated by means of software agents; it was also used to show that stakeholders can be assisted by personal software agents, since one agent can act on the behalf of a stakeholder, playing its role and following the established plan.

## References

1. Baeza-Yates, R., Ribeiro Neto, B.: Modern Information Retrieval. ACM Press, New York (1999)
2. Carmel, E.: Global Software Teams. Prentice-Hall, Englewood Cliffs (1999)
3. Cysneiros, L.M., do Prado Leite, J.C.S.: Nonfunctional Requirements: From Elicitation to Conceptual Models. IEEE Transactions on Software Engineering 30(5), 328–350 (2004)
4. Darimont, R., Lamsweerde, A.v.: Formal Refinement Patterns for Goal-Driven Requirements Elaboration. In: Proceedings 4th ACM Symposium on the Foundations of Software Engineering (FSE4), San Francisco, October 1996, pp. 179–190 (1996)
5. Dellen, B., Maurer, F.: Integrating Planning and Execution in Software Development Process. In: Proceedings of WET ICE 1996 Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp. 170–176 (1996)

6. Dhavachelvan, P., Uma, G.V.: Reliability Enhancement in Software Testing- An Agent-Based Approach for Complex Systems. In: Das, G., Gulati, V.P. (eds.) CIT 2004. LNCS, vol. 3356, pp. 282–291. Springer, Heidelberg (2004)
7. Evans, R., et al.: MESSAGE: Methodology for Engineering Systems of Software Agents, Technical Report P907, EURESCOM (2001)
8. Faltings, B.: Intelligent Agents: Software Technology for the new Millennium. In: Informatik, vol. 1 (2000)
9. Ferber, J.: Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Addison Wesley, Reading (2000)
10. Gaeta, M., Ritrovato, P.: Generalized Environment for Process Management in Cooperative Software Engineering. In: Proceedings of 26th Annual International Computer Software and Applications Conference (COMPSAC 2002) (2002)
11. Grundy, J., Ding, G., Hosking, J.: Deployed software component testing using dynamic validation agents. The Journal of Systems and Software 74, 5–14 (2005)
12. Haendchen Filho, A., Prado, H.A., Lucena, C.J.P.: A WSA-Based Architecture for Building Multi-Agent Systems. In: International Conference on Autonomous Agents (AAMAS 2007), May 2007, pp. 906–908. ACM Press, Hawaii (2007)
13. Himmelspach, J., Rohl, M., Uhrmacher, A.M.: Simulation for testing software agents - an exploration based on James. In: Proceedings of Simulation Conference, pp. 799–807 (2003)
14. Jennings, N.R., Wooldridge, M.J.: Agent-Oriented Software Engineering. In: Bradshaw, J. (ed.) Handbook of Agent Technology. AAI/MIT Press (2001)
15. Kendall, E.A.: Role Modeling for Agent System Analysis, Design, and Implementation. In: Proceedings of First International Symposium on Agent Systems and Applications Third International Symposium on Mobile Agents, p. 204 (1999)
16. Manning, C.D., Schütze, H.: Foundations of statistical natural language processing, p. 680. MIT Press, Cambridge (1999)
17. Sayão, M.: Verificação e Validação em Requisitos: Processamento da Linguagem Natural e Agentes. PUC-Rio, Rio de Janeiro, RJ. Ph.D Thesis (in portuguese)
18. Wooldridge, M.: Intelligent Agents. In: Weiss, G. (ed.) Multiagent Systems - A Modern Approach, pp. 27–77. MIT Press, Cambridge (1999)
19. Wooldridge, M., Jennings, N., Kinny, D.: The Gaia Methodology for Agent-Oriented Analysis and Design. In: Proceedings of the 3rd Int. Conference on Autonomous Agents, Seattle, WA, pp. 27–42 (1999)
20. Yu, E.: Agent-Oriented Modeling: Software versus the World. In: Wooldridge, M.J., Weiß, G., Ciancarini, P. (eds.) AOSE 2001. LNCS, vol. 2222. Springer, Heidelberg (2002)

# Integrating Business Domain Ontologies with Early Requirements Modelling

Frederik Gailly[1], Sergio España[2], Geert Poels[1], and Oscar Pastor[2]

[1] Faculty of Economics and Business Administration, Ghent University
{frederik.gailly,geert.poels}@ugent.be
[2] Centro de Investigación en Métodos de Producción de Software,
Universidad Politécnica de Valencia
{sergio.espana,opastor}@pros.upv.es

**Abstract.** Requirements engineering is an important and complex phase during systems development because it combines the description of the system's domain with the elicitation, specification and validation of application-specific requirements. In this paper we propose an ontology-based requirements engineering approach that integrates the use of domain ontologies to capture domain knowledge into early requirements modelling techniques, which primarily aim at eliciting and representing the organisational and intentional context of the system. The proposed approach can be used with different types of domain ontologies and different requirements engineering techniques. The approach is illustrated using the REA enterprise ontology, the $E^3$-value ontology and the i* requirements modelling framework.

## 1 Introduction

Early requirements engineering (RE) techniques focus on developing 'purposeful' information systems, i.e. systems that fulfil a certain purpose in an organisation [1]. Different early RE techniques with different types of orientation, like goal-oriented (e.g. i* [2]), use case-oriented (e.g. TRADE [3]) and business process-oriented (e.g. BPMN [4]) have been proposed. These approaches have been successfully used in different domains and provide concepts to develop models that include both descriptive domain knowledge and prescriptive requirements for the system. However, these techniques fail in providing a modelling approach which clearly differentiates these two perspectives, which makes it hard for domain experts to develop early requirements models [5].

The effectiveness of early RE techniques can be improved by using domain ontologies during early requirement analysis. This way, the modeller is supported in managing model granularity and assuring semantic quality. In this paper a mechanism is proposed that integrates domain knowledge in domain-independent RE languages. Our approach is based on the distinction between two types of model instantiation: linguistic instantiation and ontological instantiation. The former is concerned with the language definition of the modelling elements whereas the latter is used to classify the elements in the domain of interest [6-8].

The proposed approach differs from the use of a domain-specific language (DSL) that is derived from a domain ontology because it does not 'promote' ontology

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 282–291, 2008.

elements to modelling primitives. The result of the DSL approach is a new modelling language, whereas our approach integrates the use of domain ontologies into existing languages. We could say that the domain ontology is used at run-time during the analysis and development of the new system rather than considering it as yet another model to be included during the development process.

The proposed ontology-based early RE approach results in models with clear domain semantics. By annotating models with domain concepts, domain-specific constraints specified in the domain ontology can be used for the analysis and synthesis of early requirements models. The use of a domain ontology during the development of an early requirements model forces the modeller to take these domain-specific constraints into account.

Section 2 describes how the ontology-based modelling approach uses linguistic instantiation and ontological instantiation to integrate domain ontologies ($E^3$-value and REA) and early RE modelling languages (i*). In section 3 the approach is illustrated by using an ontology-based i* profile to annotate the goal models of an online bookstore model. Section 4 summarises the paper and discusses future work.

## 2 Ontology-Based Early RE Approach

Some researchers have used ontologies to improve the early RE phase [9-12]. Some of these efforts use a core ontology to evaluate the constructs of an existing modelling language in order to assign ontological semantics to its modelling primitives. Others have transformed domain ontologies into new domain-specific modelling languages.

The research of Kaeya and Saeki [10] and Shiboaka et al [13] is most related to our work because they annotate early requirements statements with concepts from a domain ontology that is specifically constructed to support requirements analysis in the domain of interest. After establishing this mapping the inference rules of the ontology are used to detect incompleteness and inconsistency in the requirements document. In [13] this ontology-driven requirements elicitation method is used in combination with a goal-oriented modelling technique. The resulting goal-oriented and ontology-driven requirements elicitation method improves the goal decomposition process and provides a method to integrate domain-specific knowledge in goal models. Both studies show that the repeated use of domain ontologies improves the quality of the requirements specifications.

The approach followed in this paper is different because we focus on integrating *existing* domain ontologies with no predefined structure into early requirements analysis. Additionally we link our ontology-based early RE approach to generally accepted modelling frameworks like UML and the Model-Driven Architecture.

### 2.1 Linguistic Versus Ontological Instantiation

Although not directly supported by the UML 2 infrastructure specification [14], it is recognised that two different types of "instance-of" relationships exist [8, 15]. Figure 1 represents a simple example which illustrates these two types of instantiation using the $E^3$-value ontology [9] and a simplified i* meta-model. The elements

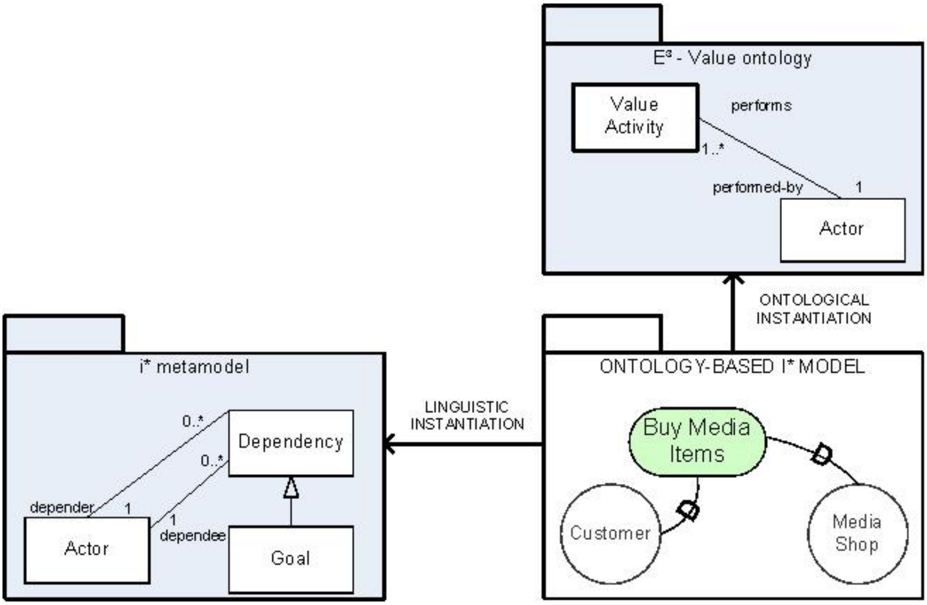



**Fig. 1.** Linguistic and ontological instantiation example

of an ontology-based i* model can be classified in two different dimensions: a linguistic dimension and an ontological dimension. The linguistic instantiation is concerned with language definitions defining the structure and presentation of the model elements (e.g. Buy Media Items is a Goal). The ontological dimension relates the model elements to domain concepts (e.g. Buy Media Items is a Value Activity, meaning a collection of operational activities performed by one actor and increasing economic value for that actor).

The distinction between these two instantiation dimensions (linguistic and ontological) provides the starting point for our ontology-based RE approach. The main idea is that domain ontologies, which represent domain knowledge, are used for ontological instantiation. In addition, traditional requirement modelling languages are used for linguistic instantiation.

This ontology-based RE approach can be integrated within the traditional Model-Driven Development paradigm, since UML 2 and MOF 2 constitute a framework that provides two types of extension mechanisms (meta-model and profile extension mechanism). In the proposed approach we distinguish between linguistic instantiation and ontological instantiation by exclusively reserving the UML meta-modelling extension mechanism for the linguistic instantiation and the profiling extension mechanism for ontological instantiation. The UML 2 infrastructure specification indicates that no general rules exist that indicate when to use a meta-model or a profile. Historically, however, profiles have been used for adapting an existing meta-model with constructs that are specific to a particular domain.

## 2.2 Ontology-Based i* Modelling

In this paper we illustrate the proposed ontology-based early requirements modelling approach by using i* goal modelling [2] for the linguistic instantiation and the $E^3$-value [9] and REA enterprise [16] ontologies for the ontological instantiation.

The **i* modelling framework** contains two basic models for the elicitation of the early requirements of the systems: the strategic dependency (SD) model and the strategic rationale (SR) model. The SD model describes the dependency relationships that exist among the organisational actors. The SR model provides an intentional description of the processes that are internal to these actors. It is commonly accepted that the i* language provides many degrees of freedom, what hinders its systematic use [17]. Several sets of guidelines have been proposed to assist modellers in the use of i*. For instance, [18] proposes a set of guidelines for architectural modelling with i*. As another example, the PRiM method offers prescriptive guidelines that address information systems modelling with i* [19]. We show that domain ontologies can also contribute to a more precise i* usage.

The type of domain ontology used during the development of a new system depends on the domain in which the system will operate. We have chosen for the illustration two complementary business domain ontologies: $E^3$-value and REA. The **$E^3$-value ontology** [9] defines a series of concepts that can be used during the value-based requirements engineering of innovative e-commerce systems. The value viewpoint of the $E^3$-value resembles the i* SD-model and as a result $E^3$-value concepts are used to annotate the elements of an SD-model. As an event ontology, **the REA enterprise ontology** [16] focuses on the events occurring within the realm of a company, their participating agents, affected resources, and regulating policies. REA can be used as a reference for modelling a single business cycle (e.g. sales-collection) or a chain of business cycles, connected through resource flows. As the i* SR-model also

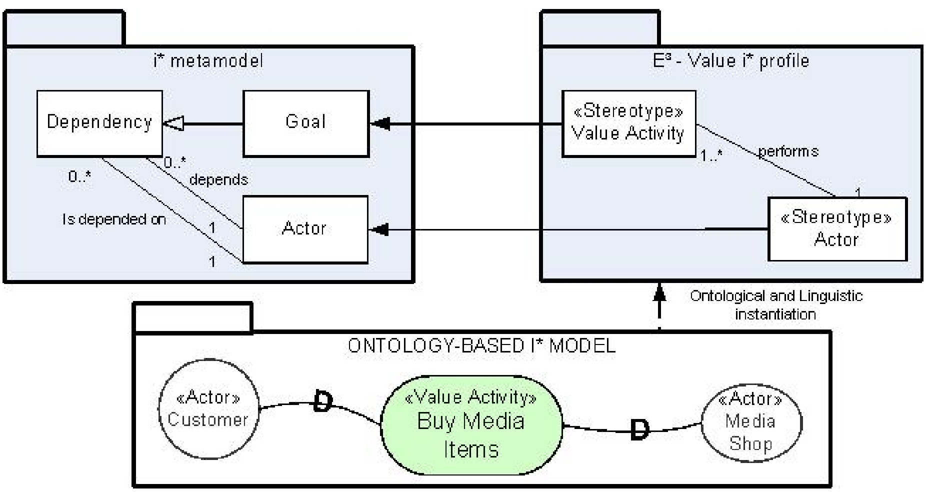



**Fig. 2.** Annotating an i* SD model with E3-value concepts

takes the perspective of one of the actors and focuses also on the internal processes that result in the achievement of this actor's goals, REA will be used for annotating the elements of the SR-model.

We followed the latest UML 2.1.2 specification [14] for the definition of $E^3$-value and REA-based i* profiles. These profiles are defined at the M2 level and use the extension relation to add business domain-specific concepts to the existing i* metamodel specified in MOF. Figure 2 shows part of the $E^3$-value profile for i*, extending the i* metamodel. The i* SD model element "Buy Media Items" is an ontological instantiation of the stereotype Value Activity defined in this profile. As Value Activity extends the metaclass Goal in the i* metamodel, "Buy Media Items" is also a linguistic instantiation of the i* Goal concept.

## 3 Illustration of the Proposal

Figures 3 and 4 present the i* models of an online bookstore such as Amazon.com. We use this example to demonstrate that integrating a domain ontology by means of ontological instantiation improves the effectiveness of the i* modelling approach and provides additional possibilities to validate the model.

### 3.1 $E^3$-Value Based SD Model of the Online Bookstore

Figure 3 presents the SD model of the order and sales process of an online bookstore, using $E^3$-value for the ontological instantiation. For the representation of the ontology-based requirements model the concrete syntax of the i* modelling language is used. The model elements are annotated by keywords in guillemets referring to business domain concepts defined in the $E^3$-value profile for i*.

We assume the following goals: the customer wants to acquire a book and the bookstore wants to generate profit by selling books. At this stage we do not focus on one specific actor but focus on the dependencies between the different stakeholders. In the context of business, we propose to relate goals to the creation of value. As a result, goals can be ontologically classified as $E^3$-value Value Activities. Following $E^3$-value, a Value Activity is connected to a Value Interface which consist of In and Out Value Ports that are connected to the ports of another Actor (of opposing economic interest) by means of Value Exchanges. In the i* SD model of the example, this means that each Value Activity needs (at least) two oppositely directed Value Exchange tasks. For instance, the "Generate income" Value Activity requires one Value Exchange task connecting the In Value Port of the bookshop Actor to the Out Value Port of the customer actor (i.e. the "Pay book" task) and one Value Exchange task connecting the In Value Port of the customer with the Out Value Port of the bookshop (i.e. the "Sell book" task). The i* task dependencies are directed from the In Value Port to the Out Value Port. Note that the same tasks are needed by the "Acquire book" Value Activity, indicating that these tasks fulfil both actors' goals.

According to $E^3$-value, Value Exchanges result in the exchange of Value Objects, which are modelled in the example as resource dependencies. In the model, "Money" and "Book" are the resources being exchanged.

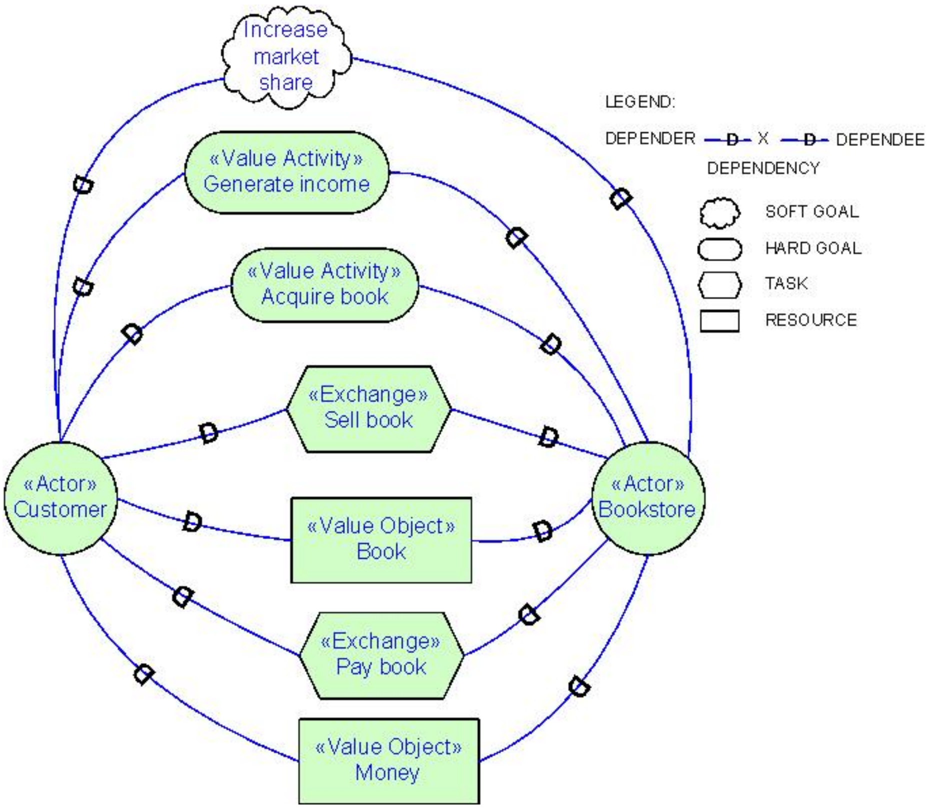



**Fig. 3.** E3-value based i* SD model of an online bookstore

The example shows that the $E^3$-value ontology offers guidelines for developing i* SD models for the business domain. For instance, by indicating that a goal is a Value Activity, we know that tasks are needed for the oppositely directed Value Exchanges that are needed by this Value Activity. Such modelling guidelines can be derived from the constraints that are part of the $E^3$-value ontology. They can be explicitly incorporated in the domain ontology-based i* profiles by adding Object Constraint Language (OCL) constraints to the domain-specific stereotypes.

### 3.2 REA Based SR Model of the Online Bookstore

The SR model (see figure 4) for the bookstore provides an intentional description of the bookstore's internal processes. The ontological overlap between $E^3$-value and REA helps specifying an SR model which corresponds to the previously developed SD model.

The SR model also starts from the "Generate income" goal. This goal is decomposed in different subgoals and subtasks which specify how it is realised. The business operation view of the REA-ontology states that business goals can be realised by

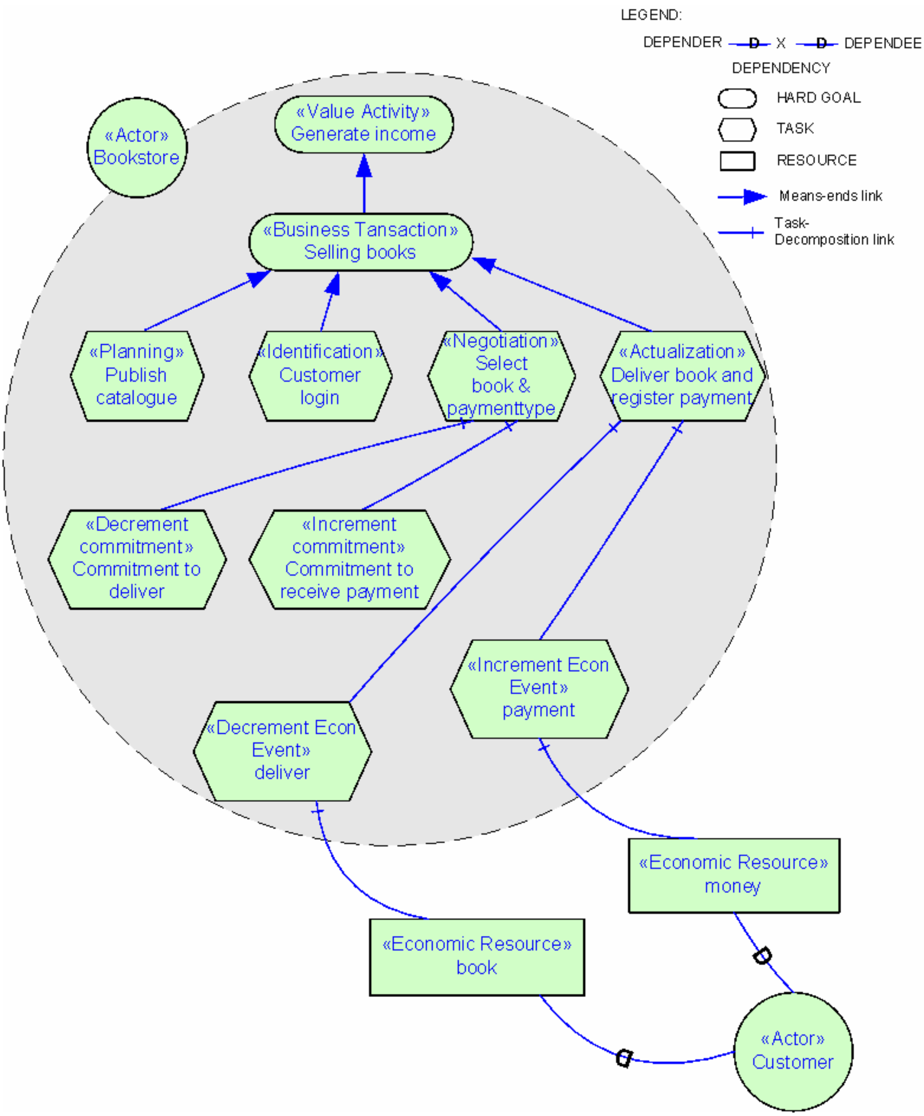



**Fig. 4.** REA based i* SR model of an online bookstore

Business Transactions which consist of four different phases: planning, identification, negotiation and actualisation. These different phases can be modelled in i* by means of tasks which represent the means to realise the Business Transaction goal “Selling books”.

The first two phases could be further decomposed into different business events but the REA ontology does not propose a real classification of these business events at this stage. Because of the accounting background of REA, the negotiation and actualisation phase are described in more detail. The negotiation phase will result in an economic bundle of commitments that can be specified in a contract. Analogous to $E^3$-value, REA describes that a decrement commitment which will result in the loss of economic resources must always be accompanied with an increment commitment that results in an increase of economic resources. In the example, the bookstore commits itself to deliver the book and the customer commits to pay for the book. These commitments are fulfilled during the actualization phase of the business transaction and are represented as tasks linked to the actualization task by means of a task decomposition link. When the transaction is actualised, the rights to use the book are transferred to the customer and the money paid for the book is available for the bookstore. The "deliver" and "payment" Economic Events are connected to the Economic Resources that are needed to realise the transaction.

The use of REA enhances the ontological correctness of i* SR models for the business domain by providing modelling guidelines. For instance, REA specifies that a business transaction consists of four different phases. Furthermore it makes sure that the modeller makes a distinction between negotiation tasks which result in economic commitments and actualisation tasks which result in the actual exchange of resources. Moreover, the traceability of the SR model to the SD model is also improved due to the ontological overlap between $E^3$-value and REA ontologies.

## 4 Conclusion and Future Research

In this paper an ontology-based early requirements modelling approach is proposed which provides a way to integrate the domain knowledge captured in a domain ontology into early requirements models. The proposed approach is based on the distinction between two types of instantiation: linguistic instantiation and ontological instantiation. For the practical implementation of the approach we propose the use of the metamodel extension mechanism for the linguistic instantiation and the profile extension mechanism for the ontological instantiation. This new approach is illustrated by extending the i* metamodels of the SD model and the SR model with respectively an $E^3$-value based profile and an REA based profile. Both $E^3$-value and REA are business domain ontologies.

Using a domain ontology for the ontological instantiation offers a series of benefits for the modeller. The definitions of the different concepts of the ontology, the relations between the concepts and especially the constraints defined for these relations can be used as guidelines during the modelling process to help assuring semantic quality. The instantiation of domain ontologies in models for a same application but with different purposes will improve the traceability between these models, especially if there is ontological overlap between these ontologies.

Our research needs to be extended in different directions. At this stage the approach has not been evaluated. In the future we plan to empirically evaluate our approach in two different settings. Firstly, we plan to evaluate the domain-specific quality of existing i* models by annotating them with REA and E3-value based stereotypes. This analysis aims at demonstrating that models can be improved significantly if a domain-specific profile extension is used. Secondly, we also plan to make the E3-value and REA i* profiles available to goal modellers in order to empirically evaluate if this results in models with higher semantic quality. Both empirical settings should pay specific attention to the knowledge required of the modeller or model evaluator. Applying the domain-specific profile requires a basic knowledge of the domain ontology and it should be investigated further if the benefits of using a domain-specific profile outweigh the cost of learning the ontology.

Another direction for future research is the development of appropriate tool support for our approach. At this stage, different modelling tools (e.g. Rational Software Modeler, Eclipse with the UML 2.0 plug-in) provide an easy way to extend MOF-based metamodels with profiles. These profiles can subsequently be applied during the development of new models or they can be used to develop domain-specific modelling tools. Our future work will focus on the development of domain ontology-based profiles that extend different types of general-purpose modelling languages (e.g. i*, UML). Specific attention will be paid to the integration of domain specific constraints into the profiles which can afterwards be used for the ontological validation of the models. Implementing these constraints will only be possible if they are transformed into OCL constraints that are defined for the domain-specific stereotypes.

Finally we will investigate whether our approach, which is grounded in the UML specification, is also useful for the subsequent phases of the Model-Driven Development process (as defined in MDA). Having an ontology-based early requirements model should make it easier to develop the conceptual schema of the system under development. In future research we want to show how early RE extensions to Model-Driven Development methods (e.g. the i* extension to the OO-method [20]) can be improved by using the proposed domain ontology based requirements modelling approach.

## References

1. Rolland, C.: Capturing System Intentionality with Maps. In: Krogstie, J., Opdahl, A.L., Brinkkemper, S. (eds.) Conceptual Modelling in Information Systems Engineering, pp. 141–158. Springer, Heidelberg (2007)
2. Yu, E.: Modelling strategic relationships for process reengineering. Department of computer science. University of Toronto, Toronto (1995)
3. Insfrán, E., Pastor, O., Wieringa, R.: Requirements Engineering-Based Conceptual Modelling. Requirements Engineering 7, 61–72 (2002)
4. OMG: Business Process Modeling Notation Specification (dtc/06-01-01). OMG (2006)
5. Estrada, H., Rebollar, A.M., Pastor, O., Mylopoulos, J.: An empirical evaluation of the i* framework in a model-based software generation environment. In: Dubois, E., Pohl, K. (eds.) CAiSE 2006. LNCS, vol. 4001, pp. 513–527. Springer, Heidelberg (2006)
6. Assmann, U., Zchaler, S., Wagner, G.: Ontologies, Meta-Models, and the Model-Driven Paradigm. In: Calero, C., Ruiz, F., Piattini, M. (eds.) Ontologies for Software Engineering and Software Technology (2006)
7. Atkinson, C., Kuhne, T.: Model-Driven Development: A Metamodeling Foundation. IEEE Software, 36–41 (September/October 2003)
8. Bézivin, J., Gerbé, O.: Towards a Precise Definition of the OMG/MDA Framework. In: Automated Software Engineering (ASE 2001), San Diego, p. 273 (2001)
9. Gordijn, J.: Value based requirements engineering: Exploring innovative e-commerce ideas. Vrije Universiteit Amsterdam (2002)

10. Kaiya, H., Saeki, M.: Using Domain Ontology as Domain Knowledge for Requirements Elicitation. In: 14th International IEEE Requirements Engineering Conference. IEEE, Minneapolis/St. Paul, Minnesota (2006)
11. Etien, A., Rolland, C.: Measuring the fitness relationship. Requirements Engineering 10, 184–197 (2005)
12. Matulevicius, R., Heymans, P., Opdahl, A.L.: Ontological Analysis of KAOS Using Separation of Reference. In: EMMSAD 2006, Luxembourg (2006)
13. Shibaoka, M., Kaiya, H., Saeki, M.: GOORE: Goal-Oriented and Ontology Driven Requirements Elicitation Method. Advances in Conceptual Modeling – Foundations and Applications, 225–234 (2007)
14. OMG: OMG Unified Modeling Language (OMG UML), Infrastructure, V2.1.2. OMG (2007)
15. Atkinson, C., Kuhne, T.: Rearchitecting the UML Infrastructure. ACM Transactions on Modeling and Computer Simulation 12, 290–321 (2002)
16. Geerts, G., McCarthy, W.: An Ontological Analysis of the Economic Primitives of the Extended-REA Enterprise Information Architecture. International Journal of Accounting Information Systems 3, 1–16 (2002)
17. Grau, G., Cares, C., Franch, X., Navarrete, F.: A comparative analysis of i*-based agent-oriented modelling techniques. In: Zhang, K., Spanoudakis, G., Visaggio, G. (eds.) Eighteenth International Conference on Software Engineering & Knowledge Engineering (SEKE 2006), San Francisco, CA, USA, pp. 657–663 (2006)
18. Grau, G., Franch, X.: On the Adequacy of i* Models for Representing and Analyzing Software Architectures. In: Hainaut, J.-L., Rundensteiner, E.A., Kirchberg, M., Bertolotto, M., Brochhausen, M., Chen, Y.-P.P., Cherfi, S.S.-S., Doerr, M., Han, H., Hartmann, S., Parsons, J., Poels, G., Rolland, C., Trujillo, J., Yu, E., Zimányie, E. (eds.) ER Workshops 2007. LNCS, vol. 4802, pp. 296–305. Springer, Heidelberg (2007)
19. Grau, G., Franch, X., Maiden, N.A.M.: PRiM: An i*-based process reengineering method for information systems specification. Information and Software Technology 50, 76–100 (2008)
20. Estrada, H., Martínez, A., Pastor, Ó.: Goal-Based Business Modeling Oriented towards Late Requirements Generation. In: Song, I.-Y., Liddle, S.W., Ling, T.-W., Scheuermann, P. (eds.) ER 2003. LNCS, vol. 2813, pp. 277–290. Springer, Heidelberg (2003)

# Goal-Oriented Authoring Approach and Design of Learning Systems

Valérie Emin[1,2], Jean-Philippe Pernin[1,2], and Viviane Guéraud[1]

[1] Laboratoire Informatique de Grenoble
110 av. de la Chimie - BP 53 - 38041 Grenoble- cedex 9 - France
[2] EducTice - Institut National de Recherche Pédagogique
19 Allée de Fontenay - BP 17424 - 69347 Lyon - cedex 07- France
valerie.emin@imag.fr, jean-philippe.pernin@imag.fr,
viviane.gueraud@imag.fr

**Abstract.** Since several years, a set of researches concerns process modeling of learning situations integrating digital technologies. Educational Modelling Languages (EML) have been carried out to provide interoperable descriptions of learning scenarios. In order to generalize the use of EML, it is necessary to provide authoring environments allowing users to express their intentions and requirements. This paper presents the core concepts of ISiS (Intentions, Strategies, and interactional Situations), a conceptual framework elaborated to structure the design of learning scenarios by teachers-designers. The framework is based on a goal-oriented approach and proposes to explicitly identify intentional, strategic, tactical and operational dimensions of a scenario.

**Keywords:** technology enhanced learning, learning scenarios, authoring approach, requirements engineering, goal oriented approach.

## 1 Introduction

Since the beginning of 2000s, a set of researches in the field of Learning Design [1] concerns process modelling of learning situations integrating digital technologies. The purpose is to produce a "learning scenario" describing organization and schedule of learning situations implying various actors (student, teacher, tutor, designer, etc.). At international level, various proposals of Educational Modeling Languages (EMLs) have been carried out as IMS-LD [2] or LDL [3]. The main challenge of EMLs is to propose a neutral and shared formalism, able to express the widest range of learning situations and to be implemented more or less automatically towards specific Information's Systems (called Learning Management Systems). An EML allows the definition of relationships between learning goals, roles of staff and learners in the learning process, performed activities, environment and resources necessary to the learning situation. As noticed by IMS-LD authors themselves [4], an EML, which mainly aims expressiveness and interoperability, is not intended to be directly manipulated by human users. Specific authoring systems [5] must be provided [6] in order to allow designers to express their requirements based on their

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 292 – 301, 2008.

own business-oriented languages and shared practices, and to develop at a lower cost their own scenarios. This formulation aims two combined goals: to provide a “computable” description able to be translated in an EML and to be understood by “domain specialists” sharing common vocabulary, disciplinary knowledge and pedagogical know-how. Our authoring approach [5] aims to better take into account learning scenario designers’ requirements and “business process” dimensions of learning scenario design which are subjects of many works in Systems Engineering and Software Engineering. We have particularly focus our researches on works concerning Goal-Oriented Requirements Engineering [7] where elicitation of goals is considered as an entry point of the specification of software systems as in Rolland and Prakash MAP model [8]. Our purpose is to provide authoring tools allowing teachers-designers belonging to close communities of practice to design their scenarios expressing intentions and adopted strategies.

This paper is organized in 4 sections following this introduction. In section 2, we describe more precisely our context, our goals and the specificity of our typical audience. Section 3 describes, with an example of scenario, our conceptual framework: ISiS model (Intentions-Strategies-Interactional situations) we propose to structure the design of learning scenarios. Before concluding, section 4 proposes a short comparison between ISiS and MAP model.

## 2 Context of Research

This research, made in collaboration between Laboratoire Informatique de Grenoble and Institut National de Recherche Pedagogique, closely associates panels of teachers in charge to co-elaborate and experiment models we want to implement. This work led us to study existing practices of sharing scenarios. Some international initiatives aim to propose scenarios databases in order to favour sharing and reuse practices between teachers, such as IDLD [9]. These databases for teachers-designers, as the one proposed by french minister of education: EduBase and PrimTice list scenarios indexed with different fields depending on the domain or subject. Their descriptions are very heterogeneous: from practice narrations to more structured formalizations. Their diversity has led us to ask ourselves on the ability of these representations to be understood and shared between several practitioners.

Before describing this research, it is necessary to qualify more precisely the specific type of designers we focus on: teachers who are called to integrate digital technologies in academic context, more precisely in the French secondary educational system (pupils from 11 to 18 years). Our *teacher-designers* have a good knowledge of the domain to be taught and can be considered in a certain way as domain-specialists. They have followed a specific training to master didactical competencies (how insure appropriation of domain-specific knowledge for targeted audiences?) and pedagogical competencies (how to organize or regulate efficient learning situations?). They do not benefit of a deep training in computer science; however, they are supposed to master a certain range of basic competencies defined by a national certification. They are generally not assisted by technical specialists in charge of implementation. They have to use existing models, methods or tools in order to develop effective learning solutions, generally in a short time compatible with their job. They may belong to a teachers' community of practice, whose emergence is made easier by Internet, and which allows new possibilities of sharing and reuse between practionners. They are frequently implied as actors of the learning unit they design; this particularity may have consequences on the precision of the design where decisions can be carried out at the runtime phase.

In order to capture needs, we asked [10] a pilot group of five teachers in secondary school to find and analyse web resources that can be qualified as "learning scenarios". Their goal was to be able to reuse and integrate them in their own domains (history, economical sciences, German, English, technology). The first given task was to analyse search strategies that the various sites provided and to evaluate the relevance of the obtained results considering their own design goals. The second task was to make a synthesis and a set of suggestions which could allow to improve the design process. A set of common findings and proposals have been formulated whereas it is possible to observe some specific concerns for each subject. Firstly, there is a very strong heterogeneity of terminology concerning scenarios between subjects and within a single subject. The terms (scenario, phase, activity, session, play, act, competence, know-how, theme, etc.) are rarely clearly defined and this often complexify the appropriation of scenarios developed by others. This fact underlines the importance to dispose of a corpus of better defined concepts to generalize real practices of mutualization. Secondly, some scenario descriptions do not deal with domain concepts such as studied knowledge or programs. This lack makes it difficult to find relevant scenarios in a specific subject, and to identify the designer's intentions which aim most frequently to solve issues related to learn some particular knowledge or acquire some particular competence or know-how. Then, for a given scenario, the pedagogical or didactical strategy is not often clarified (for example, the choice of an "expositive" approach or the choice of a more participative approach like collective problem solving). This lack of elicitation frequently implies a very precise analysis of the scenario to stand out the strategy or approach which could be an important criterion of choice. Finally, teachers suggest that the design task could be facilitated by providing libraries of typical strategies, scenarios, or situations of various granularities. Each of these components should be illustrated by concrete reproducible examples.

We present in the following section the theoretical background which can help to cover these needs and on which we base our approach.

## 3 Proposition of a Conceptual Framework

All these observations lead us, not to propose an alternative solution to EMLs, but to complete them by offering models, methods and tools to sustain design and reuse by non-computer specialists. A second part of the project, not detailed here, consists in providing mechanisms to translate business-oriented models towards different EML.

### 3.1 Theoretical Background

Our research concerns the teacher-designer activity and we base our approach on a set of complementary theoretical works concerning theory of activity:

– organization of activity, proposed by Russian psychologists such as Leontiev [11], defines hierarchical levels (activity, action, operation) which allow distinguishing intentional, strategic and tactical dimensions in the activity;
– situated character of the activity, studied by Lave and Wenger [12] insists on the importance of the context in which activities take place.
– importance of routines or schemas, that represents typical solutions given to recurrent problems in specific contexts. Those features have been particularly studied by Schank and Abelson in the context of teaching activity [13].

As previously said, we also take into account the recent works concerning Business Process Engineering and Goal-Oriented Requirements Engineering and instantiate them in the particular context of learning scenario design.

For example, in MAP Model [8], concepts of goal and intention are considered as equivalent. MAP Model is described in these terms: "A map is a process model expressed in a goal driven perspective. It provides a process representation system based on a non-deterministic ordering of goals and strategies. A map is represented as a labeled directed graph with goals as nodes and strategies as edges between goals."… "A Strategy is an approach, a manner to achieve a goal".

In our experimental context, we have confronted teachers-designers with those concepts of intentions and strategies. By linking them to their regular uses, they were able to define two different articulated levels: a first didactical level dealing with domain specific knowledge and a second pedagogical level dealing with organizing learning situations. For each level, they were able to define intentions and strategies.

From this background, we have progressively co-elaborated a "business-oriented" model ISiS (Intentions, Strategies, interactional Situations) after a two-year project tightly associating teachers-designers. After evaluation of different authoring solutions in learning design [5, 6], we have chosen to develop a graphical environment ScenEdit [14] based upon ISiS model.

### 3.2 Intentions and Strategy in the Context of Learning Scenarios

We illustrate our model with an example based on a collaborative learning scenario, LearnElec Scenario dedicated to the concept of "power of a light bulb" in electricity domain at secondary school. In this scenario, the teachers' first didactical *intention* is "to destabilize" a frequently encountered "misconception" of students in electricity which is that "proximity of the battery has an influence on current intensity".

After having elicited his intention, the teacher-designer may choose the appropriate strategy he wants to use to reach this goal. In our example, the didactical intention is implemented with a specific didactical *strategy* called "scientific investigation strategy" composed of four *phases*: hypothesis elaboration, solution elaboration, hypothesis testing and conclusion as you can see on figure 1.

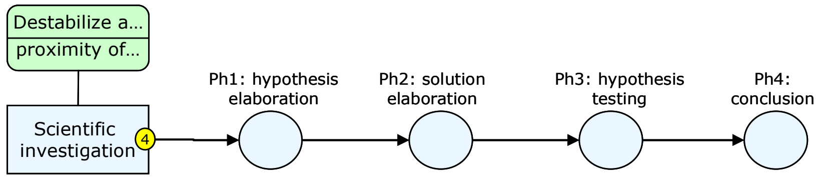



**Fig. 1.** An example of intentions and strategies elaborated by teachers in electricity

Each phase can be performed through various pedagogical modes and can be refined by another intention according to the type of activity, the availability of computer services, etc.. In our example, the first didactical phase "hypothesis elaboration" is refined by a pedagogical intention called "increase the ability to work in a collaborative way". This intention is implemented with a strategy called "elaborating a proposal by making a consensus" composed of two phases: "Make an individual proposal" and "Confront proposals. Obtain a consensus". For each phase, an *interactional situation* can be defined: "Individual proposal on a wiki" and "Argued debate on a forum with consensus". During these two phases the teacher has an activity of "Group management" symbolized by an interactional situation called "Group management".

In the following section, we present more formally ISiS conceptual model.

### 3.3 Our Proposal: ISiS Model

ISiS model is a conceptual framework elaborated to structure the design of learning scenarios by teachers and to favour sharing and reuse practices between designers. ISiS model is based on four complementary principles. Design and exchange of learning scenarios must be facilitated by:

- elicitation of context, particularly by distinguishing the knowledge context from the situational context of a learning unit;
- elicitation of intentional, strategic, tactical and operational dimensions;
- capability to provide flexible design processes allowing different combinations of steps;
- reuse of existing scenarios, components or design patterns which allow the teacher to design more efficiently his scenarios.

According to ISiS model (cf. fig. 2), organization and planning of a learning unit can be described with a high-level *structuring scenario* which reflects designer's intentional and strategic dimensions. A structuring scenario organizes the scenario in different *phases* or *cases* by means of *intentions* and *strategies*. Each phase or case can be either recursively refined by a new intention or linked at a tactical level to a suitable *interactional situation*. An interactional situation can be itself described by a more low-level *interactional scenario* which defines in an operational way the precise organization of situations (in terms of activities, interactions, roles, tools, services, provided or produced resources, etc.). The interactional scenarios are the level typically illustrated with EML examples of implementation.

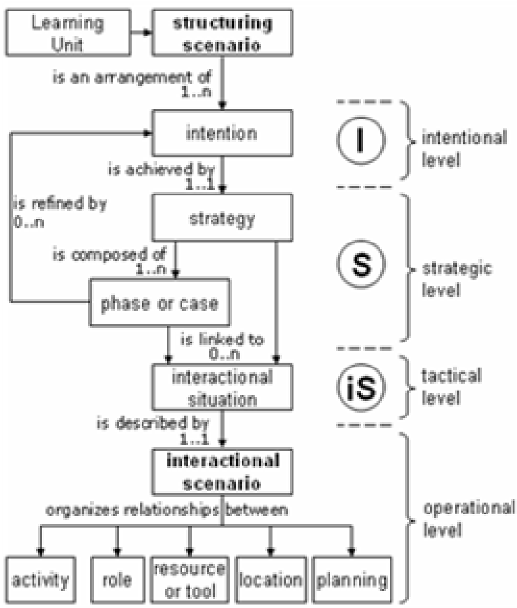



**Fig. 2.** An overview of ISiS model

Figure 2 provides an overview of the ISiS model which proposes to structure the design of a scenario describing organization and planned execution of a learning unit:

- the *I level* (Intention) allows to describe the designer's intentions. In our field, intentions are closely linked to the *knowledge context* which defines targeted knowledge items (concepts, notions, competencies, know-how, abilities, conceptions or misconceptions, etc.). Intentions for the designer can be for example to reinforce a specific competence in electricity, to favour a notion discovery, to destabilize a frequently met misconception, etc;
- the *S level* (Strategy) is related to strategic features. In order to reach goals related to the formulated intentions at *I* level, the designer opts for the strategy he considers as the most appropriate. Two main kinds of strategies can be distinguished: *sequencing strategies* organize the arrangement of logical phases (for example a scientific investigation strategy can be modeled as a series of four phases), *distribution strategies* plan different solutions for identified cases (for example, a differentiation strategy takes account of three possible levels of mastering). Strategies can be combined by successive refinements: for example, a sequencing strategy may precise one of the case of a distribution strategy;
- the *iS level* (interactional Situation) represents the tactical level, i.e. the proposed solution to implement the formulated intentions and strategies. We consider that, for a new problem, a teacher-designer doesn't rebuild from scratch a new specific solution. As underlined by works about schemas and routines in teaching activities [13], the teacher bases planning or adjustments upon a library of mastered solutions, which are triggered by specific events. In the same way, we assume that a scenario designer selects situations appropriated to his intentions and strategies,

from a library of components. Each component, an "*interactional situation*", is made up of a collection of interactions with a specific set of roles, tools, resources, according to the *situational context.* The situational context is characterized by a set of variables such as *resources* that can be manipulated to support the activities (document, tools, services), *locations* where activities can take place, *planning elements* in which activities must be scheduled or number of learners, *roles* which can be distributed. For example, in order to precise his scenario for the "solution elaboration phase" in a collaborative way and for distant learners, a designer can choose a typical situation called "argued debate on a forum with consensus". In another context, as for example for pupils in difficulty in classroom, he can choose a more individualized situation, such "choosing a solution between different possible proposals by using a MCQ tool"

– the *interactional scenario* (operational level) allows to precisely describe the details of the solution, i.e. organization and process of each interactional situation. Nowadays, EMLs essentially focus on description of this operational level by organizing relationships between actors, activities and resources, in a given language.

ISiS model proposes to explicit the upper levels (I, S and iS) that are generally not precisely defined by current methods or tools.

### 3.4 Implementation of ISiS Model: Towards Flexible and Continued Design Processes

ISiS framework is not properly a method and does not propose a specific order to combine design steps. ISiS is based on the hypothesis that all dimensions of a scenario (intentions, strategies, situations, activities, resources) must be elicited and interlinked in order to make easier design, appropriation, sharing and reuse. In our experimentations, we have analyzed the tasks led by teachers-designers [10]. Several design processes may be considered as shown by different studies we have led with teachers-designers. Some teachers may choose a top-down approach by hierarchically defining their intentions, strategies, situations, etc., while others may prefer to adopt a bottom-up approach by "rebuilding" a scenario from resources or patterns that they want to integrate. Then, one of our hypotheses is that the design of learning scenario cannot be modeled as a linear process without significantly reduce designers' creativity. According to the designer type, according to the uses within a precise community of practice, several kinds of objects or methods are shared. So, resources, pedagogical methods, typical situations could constitute an entry point from which design steps will be combined. From this entry point (for example typical interactional situations), the designer may alternatively and recursively perform design tasks. From these principles, ISiS model has been implemented successively by different kinds of tools (diagram designer or mind mapping software). In a first step, we have elaborated paper forms to express the different dimensions of the design (knowledge context, situational context, intentions, strategies, interactional situations, activities, etc.). We have also adapted mind mapping software where each node represents a concept (e.g. strategies, phases, interactional situation) and can be edited with a specific electronic form. These first tools based on ISiS model have been experimented in secondary school by a group of five teachers in technological subjects, associated to INRP institute. Each "teacher-designer" had one month to model a learning sequence that he must implement during the school year, by using provided tools. All teachers accomplished the required task in the prescribed time, and the different produced sequences had a duration varying between two and six hours. Moreover, one teacher has covered the complete process by (1) describing his scenario with the paper forms, (2) encoding the designed scenario with a specific editor (LAMS), (3) implementing the result automatically towards Moodle, a learning management system and (4) testing the scenario with his pupils. After this first experimentation, the teachers have been questioned about their design activity.

Teachers-designers answers have shown the benefits of the model to improve the quality of created scenarios, to illustrate the importance of elicitation of intentions and strategies by users themselves, to better understand the scenarios created by others and to simplify the design process by reducing distance between users requirements and the effectively implemented system.

The following points can be finally raised:

- elicitation of intentions and strategies allows the teacher-designer to better understand a scenario created by a peer;
- teachers expressed the need to provide them with reusable components allowing (a) to decrease significantly the design duration and (b) to explore solutions, proposed by peers, able to suggest a renewal of practices;
- the complete implementation on a LMS done by one of the teacher has been considered easier using ISiS model;
- provided tools (paper forms and mind mapping tool) have been considered as too costly to be integrated in a regular professional use.

These first results show the capabilities of ISiS model to encourage an efficient authoring approach. The main restriction formulated by users consists in providing adapted graphical tools. To solve this restriction, we currently develop a specific graphical authoring environment named ScenEdit [14] based on ISiS Model.

## 4 Discussion: MAP and ISiS

In the beginning of this paper we have quoted the works concerning MAP model [8] as a background of our model. We propose here to quickly describe main common features and differences between the two models. A more complete study would be necessary to really compare them with common uses cases.

MAP and ISiS are both model dedicated to design process in a goal oriented perspective. MAP is a more generic model than ISiS which is dedicated to a specific learning "business-process". MAP has been defined in order to sustain design process while ISiS aims to imply actors themselves in the design of the process. To reach that goal, it is necessary to provide users with sufficiently accessible conceptual terms. For French teachers, concepts of "pedagogical intention", "learning strategy", and "learning situation" belong to common vocabulary. It seems to be coherent to propose a specific model based on those terms. The concepts of intention and strategy in MAP and ISiS are quite close. When MAP considers a strategy as a way to link two goals, ISiS proposes to sequence two intentions where the first intention is linked to the strategy. Implicitly the model assumes that the second intention will be invoked after that the first strategy has been implemented. Concerning intentions, ISiS proposes to gather two or more intentions of different kinds in a same group. This allows to associate a same strategy to several intentions, which is an explicit demand of some teachers-designers. MAP proposes to organize strategies with different relationships such as multi-thread, bundle, path and multi-path relationships. ISiS allows those relationships, at the exception of multi-threading (for ISiS, all strategies must be exclusive). In ISiS, alternatives are represented by a specific *distribution strategy*, which allows to distinguish several sub-strategies linked to sub-intentions which refines the main one. The concept of variability [15] with MAP may be declined in ISiS according two ways: by choosing different strategy or by associating different operational solutions to a same strategy. ISiS manages a "tactical level" refining the modelization of strategies by linking them to their typical solutions.

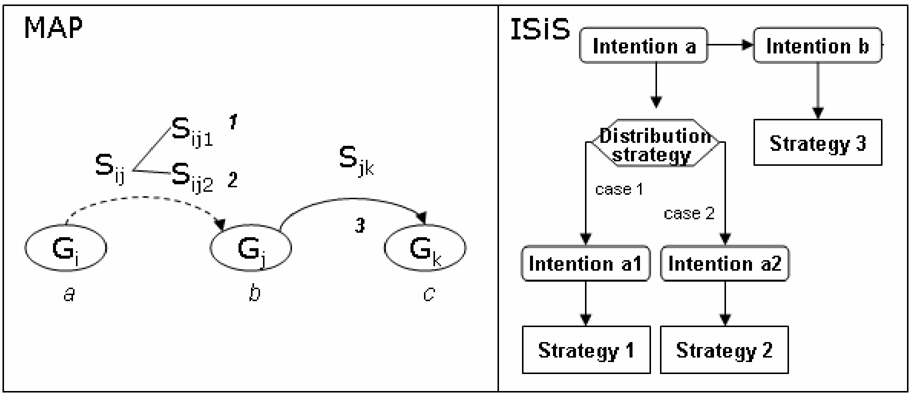



**Fig. 3.** Graphical notations with MAP and ISiS

From a graphical point of view, MAP and ISiS adopt a different approach. MAP proposes a process as a labeled directed graph with goals as nodes and strategies as edges between goals. Applied to complex example, this kind of representation may become quickly complex and require to decompose the problem by successive refinements. In order to be accessible to teachers-designers, ISiS proposes to organize a process as an oriented tree where nodes (intentions, strategies, items, solutions) belong to a defined vertical level. This may improve readability but impose some restrictions about the depth of the tree. Figure 3 shows a same example represented by MAP and ISiS graphical notations.

This comparison shows probably that the two models may be confronted in order to enrich the expressivity of intention-driven notations for processes.

## 5 Conclusion

In this paper, we have presented an overview of ISiS Model, a "business-oriented process" whose purpose is to assist teachers in the design of learning scenarios and to favour sharing and reuse practices. The model, co-elaborated with panel of users, seems to be efficient, according first experimentations. Our actual priority is to experiment the dedicated authoring environment we have developed from our conceptual model. This experimentation aims essentially to validate the visual representation of a scenario we propose and to enhance the system with patterns or components allowing new effective practices of sharing and reuse.

## References


1. Rawlings, A., van Rosmalen, P., Koper, E.J.R., Rodríguez-Artacho, M.R., Lefrere, P.: Survey of Educational Modelling Languages (EMLs). CEN/ISSS WS/Learning Technologies (2002)
2. Koper, R., Tattersall, C.: Learning Design: A Handbook on Modelling and Delivering Networked Education and Training. Springer, Heidelberg (2005)
3. Martel, C., Vignollet, L., Ferraris, C., David, J.P., Lejeune, A.: Modeling collaborative learning activities on e-learning platforms. In: Proceedings of the 6th IEEE ICALT, Kerkrade, pp. 707–709 (2006)
4. Koper, R.: Current Research in Learning Design. Educational Technology & Society 9(1), 13–22 (2006)
5. Murray, T., Blessing, S.: Authoring Tools for Advanced Technology Learning Environment. In: Toward Cost-Effective Adaptive, Interactive and Intelligent Educational Software, p. 571. Kluwer Academic Publishers, Dordrecht (2003)
6. Botturi, L., Cantoni, L., Lepori, B., Tardini, S.: Fast Prototyping as a Communication Catalyst for E-Learning Design: Making the Transition to E-Learning: Strategies and Issues. In: Hershey, M.B., Janes, D. (eds.) (2006)
7. Van Lamsweerde, A.: Goal-Oriented Requirements Engineering: A Guided Tour. In: Fifth IEEE International Symposium on Requirements Engineering (RE 2001), p. 249 (2001)
8. Rolland, C., Prakash, N., Benjamen, A.: A Multi-Model View of Process Modelling. Requirements Engineering 4(4), 169–187 (1999)
9. Lundgren-Cayrol, K., Marino, O., Paquette, G., Léonard, M., De La Teja, I.: Implementation and deployment process of IMS Learning Design: Findings from the Canadian IDLD research project. In: Proc. Conf. ICALT 2006, pp. 581–585 (2006)
10. Emin, V., Pernin, J.-P., Prieur, M., Sanchez, E.: Stratégies d'élaboration, de réutilisation et d'indexation de scénarios. actes du colloque. In: SCENARIO 2007, Montréal, pp. 25–32 (2007)
11. Leontiev, A.N.: The problem of activity in psychology. In: Wertsch, J. (ed.) The concept of activity in Soviet psychology, Armonk, Sharpe, New York (1981)
12. Lave, J., Wenger, E.: Situated Learning: Legitimate Peripheral Participation. Cambridge University Press, Cambridge (1990)
13. Schank, R.C., Abelson, R.: Scripts, plans, goals and understanding. Erlbaum, Hillsdale (1977)
14. Emin, V.: ScenEdit: an authoring environment for designing learning scenarios. In: Poster ICALT 2008, IEEE International Conference on Advanced Learning Technologies, Santander (2008)
15. Rolland, C., Prakash, N.: On the Adequate Modeling of Business Process Families. In: Workshop on Business Process Modelling, Development, and Support (BPMDS), Trondheim, Norway (June 2007)

# Timing Nonfunctional Requirements

Ivan J. Jureta and Stéphane Faulkner

PReCISE, University of Namur, Belgium
ivan.jureta@fundp.ac.be, stephane.faulkner@fundp.ac.be

**Abstract.** Analysis of temporal properties of nonfunctional – i.e., quality – requirements (NFRs) has not received significant attention. In response, this paper introduces basic concepts and techniques needed for the specification and analysis of time properties of NFRs.

## 1 Introduction

*Topic.* Software can be characterized by *what* services it provides to stakeholders, and *how well* it provides these services, i.e., dependably, rapidly, and efficiently, among other. The former are usually called *functional* and the latter *nonfunctional* (i.e., quality) requirements [2]. Research and industry seem to agree that several complementary approaches spanning various steps of the software development process and software evaluation are needed to ensure a degree of quality as close as feasible to the ideal levels desired by the stakeholders [3]. Assistance can be provided to the software engineer in the form of dedicated concepts and techniques for the specification and analysis of software nonfunctional requirements. Currently, such help relies to a considerable extent on the theoretical foundations laid out in the established Nonfunctional Requirements (NFR) framework [2] which combines the concept of nonfunctional requirement for representation, with contribution relationships for reasoning about NFRs.

*Problem and Contributions.* One perspective on setting quality levels and assuring that they are met, which has remained outside the scope of the NFR framework is the specification and analysis of how levels of satisfaction of NFRs vary over time. The framework does not cover the definition and analysis of properties of time intervals over which a particular sastisfaction level is expected of NFRs – instead, the implicit assumption seems to be that every nonfunctional requirement is to be fixed and enforced during the entire duration of software operation or the development and evaluation project. Change of environment conditions, stakeholders' requirements, and conflicts between NFRs make this assumption unrealistic: there certainly are intervals in system operation, during which some NFR is to be favored over other, only to reverse the preferences at some other times. Specifying expected levels of satisfaction *over time* for NFRs helps in detecting interdependencies between NFR satisfaction levels and indicates the need for discovering conditions for the anticipated variation of satisfaction levels. This paper suggests concepts and techniques for introducing the information about time in the specification and analysis of NFRs, that is, to enable temporal properties of NFRs to be rendered explicit and studied.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 302–311, 2008.

*Running example.* To illustrate the suggested concepts and techniques, initial requirements expressed for Air-Traffic Management (ATM) information systems were analyzed. Between 1994 and 1998, the European Organization for the Safety of Air Navigation (Eurocontrol) organized a number of consultation workshops for the aviation community in the aim of documenting these stakeholders' expectations regarding the ATM information systems strategy for the period after the year 2000. Information in workshop deliverables [1] contains actual initial considerations expressed by the various stakeholders, including a large number of quality-related concerns. In the present paper, fragments from the case study are used to exemplify concepts and techniques introduced in the remainder.

## 2 Timeline, Condition and Policy Concepts

Introducing time in the analysis of nonfunctional concerns changes how the software engineer proceeds to the selection of alternative combinations of satisfaction levels, which in essence amounts to the selection of a particular set of desired software properties among alternatives. Without considering the variation of NFR satisfaction over time, the software engineer selects among static representations of software alternatives. However, when time is accounted for, the software engineer can compare the alternative sets of behaviors of the software in relation to NFRs over time. Static representations of alternatives such as, e.g., NFR goal contribution graphs ([2] and related) cannot provide such comparisons.

Overall, adding time information to an NFR proceeds as follows: A *timeline*, composed of *time points* organized into *intervals* is defined for each NFR; *conditions* for the variation of desired NFR satisfaction levels are given along with *policies* (see below) to apply for ensuring the desired quality leves be met; a condition starts an interval, and the interval is set to last until the change of conditions (timing out can also be a condition, so that specific durations can be accounted for) – the satisfaction level of the NFR is thus assumed stable accross each interval. Timelines of different NFRs are then combined until the entire NFR specification is accounted for in the *general NFR timeline*. As iterations in specification due to incremental acquisition of NFR-related knowledge both by the stakeholders and the software engineer make distributed and computer-assisted NFR specification more likely than a sequential process executed exclusively by the software engineer, it is not appropriate to work on the NFR book timeline without having elaborated individual NFR timelines. Combining timelines is partly automated through the timeline synchronization technique (see, Technique 3) algorithm that generates alternative paths based on tradeoffs between satisfaction levels of distinct NFRs. From there on, selecting among alternative timelines is a matter of analysis of the general NFR timeline using techniques described below (§3).

The presence of labels make the timeline a particularly useful means for relating within a single representation the conditions, policies, stakeholders, along with time information, and metric values. This integration, along with timeline merging, and creation and comparison of alternatives are discussed later on

(§3), after the present discussion defines basic timeline conceptualization and the subsequent one introduces the condition and policy concepts.

**Definition 1.** *A* ***timeline*** $\mathcal{T}_q \stackrel{def}{=} (G_q, F_q)$ *for the NFR* $q \in \mathcal{Q}$ *consists of:*

1. *A graph* $G_q = (N_q, E_q, source_q, target_q)$ *defined over a set of nodes* $N_q$ *and multiset (i.e., bag) of edges, with* $source_q : E_q \longrightarrow N_q$ *and* $target_q : E_q \longrightarrow N_q$*, which map, resp., an edge to its source node and an edge to its target node.*
2. *A set of functions* $F_q$ *used to label the graph* $G_q$*, as follows:*
   (a) $timing_q : N_q \longmapsto \mathbb{T}$ *is a partial function mapping edges to time points, whenever times can be specified, i.e., are known by the stakeholders or the engineer. Intervals can be derived for successor edges, and interval duration by substituting the left from right endpoint, provided the two are known. The time domain is introduced in Def.2.*
   (b) $condition_q : E_q \longmapsto \mathbb{P}(\mathcal{V}_{\mathsf{Condition}})$ *places sets of conditions on edges of* $G_q$.
   (c) $policy_q : E_q \longmapsto \mathbb{P}(\mathcal{V}_{\mathsf{Policy}})$ *labels edges in* $G_q$ *with sets of policies.*
   (d) $agreedBy_q : E_q \longmapsto \mathbb{P}(\bigcup_{q \in \mathcal{Q}} q.\mathsf{Agent.Name})$ *maps an edge to a set of agents specified in the* Agent *attribute of the given NFR* $q$*. The agents are those who explicitly marked agreement with the valued metrics specified for the given edge (§3).*
   (e) $setMetric_q : E_q \longmapsto \mathbb{P}(\mathcal{M} \times \mathcal{V}_{\mathsf{Metric}})$ *associates sets of valued metrics to edges of the timeline graph.*

**Definition 2.** *The* ***time domain*** $(\mathbb{T}, \leq_t)$ *is a set of* ***time points*** $\mathbb{T}$ *ordered by the binary relation* $\leq_t$*. As usual,* $t_1 <_t t_2$ *iff* $t_1 \leq_t t_2$ *and* $t_1 \neq_t t_2$*. The relation* $\leq_t$ *is linear, i.e., if* $t_1 \leq_t t_2$ *and* $t_2 \leq_t t_1$ *then* $t_1 =_t t_2$ *(antisymmetry), if* $t_1 \leq_t t_2$ *and* $t_2 \leq_t t_3$ *then* $t_1 \leq_t t_3$ *(transitivity), and* $\forall t_1, t_2 \in \mathbb{T}, t_1 \leq_t t_2$ *or* $t_2 \leq_t t_1$ *(totality—any two members of* $\mathbb{T}$ *are comparable under the relation* $\leq_t$*).*

**Definition 3.** *An* ***interval*** $I$ *is a convex subset of* $\mathbb{T}$.

*Notational conventions.* Right endpoint of an interval $I$, denoted $r(I)$, is its supremum (also called its greatest upper bound); left endpoint, $l(I)$, of $I$ is its infimum (least upper bound). $I = (t_i, t_j]$ is a left-open (i.e., $t_i \notin I$) and right-closed ($t_j \in I$) interval.

When applied, the chosen conceptualization gives a graph in which labels indicate (i) the conditions and policies provided and accepted by (ii) a group of stakeholders for setting (iii) specific target values for metrics that measure the degree of satisfaction of NFRs expected to be satisfied over the given interval. An important property of the timeline is that it is not linear: it branches, so that distinct edges can share same nodes. Timelines branch because of trade-offs between NFRs that cannot be satisfied simultaneously to ideal levels or because of change in conditions that require revision of target metric levels. However, knowing that target values must change with variation of target values of other metrics is helpful to a limited extent: in addition to variation, both the conditions under which variation is expected to occur and the policies to adopt in the aim of affecting the expected degree of the metrics are of interest. Consequently, each interval in the timeline is labeled with the Condition and Policy attributes.

**Definition 4.** *A **condition** describes constraints on the context or part thereof for the NFR (for which that timeline is being built), under which a given set of metric target values is considered desirable by a set of stakeholders.*

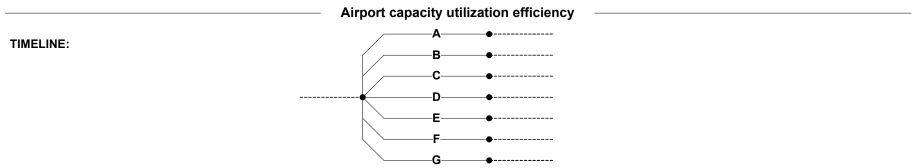
Knowing the conditions facilitates the identification of actions which will aim at changing or maintaining the given conditions. In other words, the content of Policy refers to the goals, processes, tasks, or actions whose achievement or execution leads, for the given interval, to the balacing of metric target values visible on the timeline for that interval.

**Definition 5.** *A **policy** is a goal, process, task, action, or a set thereof that the stakeholders believe needs to be achieved (for goals) or executed in the aim of ensuring the metric target values be reached under the condition associated to the policy in the given timeline.*

## 3 Using Timelines, Conditions and Policies

Information on time, conditions, and policies is closely related: analyzing one without the others is possible, but additional techniques can be applied if these considerations are treated together within a single cadre. Herein, two techniques, *defensive NFR planning* and *NFR planning* are presented. Each involves the acquisition of time and condition/policy information and its combination in order to arrive at a timeline for the given NFR. After timelines are acquired, they are then combined first locally within the same context (i.e., meaningfully related set of NFRs, such as the NFRs obtained by decomposing some higher-level NFR), and then accross contexts using in both cases the timeline synchronization technique (Technique 3). As the said technique generates a large number of alternative paths on synchronized timelines, pruning techniques are also suggested to assist in selecting the most desirable one on which the stakeholders can agree. The techniques presented in the remainder of the section thus integrate the various information acquired for the time and condition/policy considerations. Notice however, that timelines not only integrate this information, but also part of preferences information and record agreement between the stakeholders participating in the development process: preferences appear in the form of metrics and branches which result from alternative sets of metric values, themselves the result either of varying conditions, or disagreement of the stakeholders—those agreeing on labels for timeline edges are specified in the labeled timeline. Elicitation and analysis of preferences and agreement is relegated to later sections for clarity of presentation.

**Technique 1.** *(Defensive NFR planning.)* Defensive planning serves as a first step in making the software and the relevant part of its environment responsive to random events whose occurrence can affect the level of NFR satisfaction. Because of inherent unpredictability, the duration of such events and hence the starting and ending time are unlikely to be available. The construction of the timeline is condition-driven, whereby combinations of conditions are identified,




| *Label:* | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| **DURATION:** | *varies* | *varies* | *varies* | *varies* | *varies* | *varies* | *varies* |
| **CONDITION:** | Normal conditions | Light weather contingency | Heavy weather contingency | Short term demand changes AND Normal conditions | Light weather contingency | Heavy weather contingency | Heavy weather contingency |
| **POLICY:** | *No specific policy - usual operation* | Safety before capacity, Redirect to regional airports | Safety before capacity, Keep on ground, Redirect early to regional airports | Maintain average delay belows 35min | Safety before capacity, Redirect to regional airports | Safety before capacity, Keep on ground, Redirect early to regional airports | Safety before capacity, Keep on ground, Redirect early to regional airports, Avoid PopAreaZones |
| **NFC & METRIC:** | | | | | | | |
| Airport capacity utilization efficiency | | | | | | | |
| ↑ **AirportCapUtil** | 75% | 55% | 30% | 68% | 45% | 30% | 30% |
| Safety under weather contingency | | | | | | | |
| ↓ **AirborneWaitingCont** | | 0.12 | 0.04 | | 0.12 | 0.04 | 0.045 |
| Flexibility under short-term changes in demand | | | | | | | |
| ↓ **STChangesDelay** | | | | 2.15 | 2.27 | 2.27 | 2.27 |
| ↓ **SlotChangeDelay** | | | | 0.11 | 0.12 | 0.12 | 0.12 |
| Environment noise pollution | | | | | | | |
| ↓ **T&LOverPopAreaZones** | 0% | 5% | 20% | 5% | 7% | 30% | 20% |
| **AGREED BY** | Airport representatives, Local city authorities | Weather Service Providers (WSP), Airport representatives, Local city authorities | Weather Service Providers (WSP), Airport representatives, Local city authorities | Airport representatives, Airline representatives, Local city authorities | Airport representatives, Airline representatives, Weather Service Providers (WSP), Local city authorities | Airport representatives, Airline representatives, Weather Service Providers (WSP) | Local city authorities |

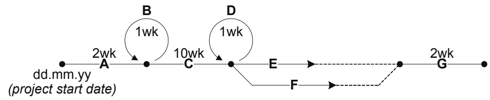
**Fig. 1.** Defensive NFR planning for the NFR Airport capacity utilization efficiency

desired (target) metric values are given for all metrics of NFR that interact with the NFR for which the timeline is being built, stakeholders agreeing with the set metric values are identified and noted, alternative values are written down as well along with the stakeholders who suggested them, and finally, policies for meeting the given quality levels are referenced on the obtained timeline. In case an alternative is suggested, the condition line branches, as in Fig. 1. In essence, timelines used for defensive NFR planning are decision trees which provide an overview of scenarios (i.e., sets of conditions) and associate with each a set of stakeholders, policies, and target metric values. A simple way to visualize such a timeline is with a table, as in Fig. 1, where: conditions are noted above columns in which corresponding target metric values, agreed upon by a set of referenced stakeholders, are set; the bottom row identifies the policies the stakeholders suggested to meet the target quality levels.

*Example 1.* Fig. 1 is an example of the output obtained by application of Technique 1. The graph indicates that all columns in the table are alternatives. Metric values are examples only. The metrics to list in the table are those that correlate with the metrics of the NFR for which the timeline is being built (the latter

**Expenditure visibility**

**TIMELINE:**




| *Label:* | **A** | **B** | **C** | **D** | **E** | **F** | **G** |
|---|---|---|---|---|---|---|---|
| **DURATION:** | 2 weeks | 1 week | 10 weeks | 1 week | Until project end | Until project end | 2 weeks |
| **CONDITION:** | Project start, Cost control team new to EMA | Unacceptably many errors in expenditure database | Error count acceptable in the EMA expenditure database | Significant discrepancies between forecast and actual exp. | | Minor discrepancies between forecast and real expenditures | Project end announced, expenditure analysis and summary requested |
| **POLICY:** | Train Cost control team for quick exp. registration and early announc. | Train Cost control team to balance speed and accuracy | Train Cost control team to anticipate and forecast expenditures, and report these | Revise forecast, detect forecasting problems do more training for the Cost control team | Hire external expert to manage forecasting effort | Maintain recording, announc., and forecast performance | Stop recording, announc., and forecasting, produce analysis of all exp. |
| **NFC & METRIC:** | | | | | | | |
| Expenditure visibility | | | | | | | |
| ↓ **TimeToRecExp** | 8h | 5h | 3h | 3h | 2h | 2h | |
| ↑ **TimeTillExpAnn** | 1 day | 3 days | 4 days | 4 days | 5 days | 5 days | |
| Expenditure database accuracy | | | | | | | |
| ↓ **ProbErroneousEntry** | High | High | Low | Low | Low | Low | |
| Expenditure forecast | | | | | | | |
| ↓ **ProbErroneousForecast** | | | High | High | Low | Low | |
| Budget utilization | | | | | | | |
| ↓ **ProbBudgetOverrun** | Med | Med | Med | Med | Low-Med | Low | |
| **AGREED BY** | PM, Accounting team, Cost control training team | PM, Accounting team, Cost control training team | PM, Accounting team, Cost control training team | PM, External forecast expert, Accounting team, Cost control training team | PM, Accounting team, Cost control training team, External forecast expert | PM, Accounting team, Cost control training team | PM, Accounting team, Cost control training team |

**Fig. 2.** NFR planning with a timeline for the NFR Expenditure visibility

are included in the table as well). Each column in the table contains the conditions, target metric values desired under the given conditions, the stakeholders who agree on these metric values, and the policies whose application is expected to lead to set metric levels. As discussed above, another visualization of the same information is a decision tree in which leaves indicate conditions, policies, stakeholders, and metric values, while branching occurs whenever conditions are combined. Without going into the detail of NFRs for the NFR listed in Fig. 1, the metrics mentioned therein are specific to the ATM domain—for instance, AirportCapUtil compares the estimated or real capacity utilization with theoretical airport capacity, whereas T&LOverPopAreaZones may either compare levels of noise in noise sensitive areas (i.e., populated areas) over which takeoff and landing may be permitted under specific conditions, or the frequency of takeoff and landing over these areas. Notice that an arrow, oriented up for metrics to maximize, and down for those to minimize is also mentioned in the table. Observe further that combinations of conditions are indicated as for Short term demand changes, and that alternative sets of metrics, stakeholders, or policies are indicated by splitting the condition line, as for Heavy weather contingency when combined with Short term demand changes.

**Technique 2.** *(NFR planning.)* The NFR planning technique is applied instead of Defensive NFR planning (Technique 1) if: (i) conditions are predictable (which is often the case when defining quality levels for processes common in the given domain – here, e.g., communication during landing or takeoff, various security procedures, slot changes, and so on, as well as the project management processes associated to the development or evaluation project – see, Ex.2), and (ii) sequence of conditions may matter (in Defensive nfr planning, sequence is difficult to establish because of unpredictability). The graph representation of the timeline tends then to differ in NFR planning in that the timeline does not involve alternatives only (or mostly) as is the case in Defensive NFR planning. To simplify the reading of the timeline, loops indicate edges that can be optionally taken, whereby it is assumed that the loop is executed only once if no other indication is given in the graph (such as the maximal number of times the loop can be executed).

*Example 2.* Fig.2 shows a tabular representation of timeline information and gives a graphical representation of the timeline to show alternatives in the form of branches and loops. Intervals in the timeline carry alphabetical labels to associate the two parts of the visualization.

**Technique 3.** *(Timeline synchronization.)* Synchronization amounts to combining timelines so as to build the general NFR timeline which integrates conditions and policies associated with all of the individual NRF timelines. Accomplishing this manually while accounting for the various possible alternatives is inappropriate, as the merging of two relatively simple timelines illustrates: in Fig.3, the usual representation with loops is first expanded for both timelines to show the alternative paths, while the rightmost part of the figure shows a part of the combined timeline in which many alternatives are represented. Synchronization is thus partly automated to make the task feasible; it proceeds as follows:

1. Select two timelines $T_{q_1}$ and $T_{q_2}$ in the same context.
2. Apply the **GenerateAlternatives** algorithm (see below) to $\mathcal{T}_{q_1}$ and $\mathcal{T}_{q_2}$ generate a third timeline $\mathcal{T}_{q_3}$ which contains all possible alternative paths obtained by combining the intervals from each of the two timelines.
3. Apply pruning techniques (see below) to reduce the number of relevant alternative paths in $\mathcal{T}_{q_3}$.

*Example 3.* Fig.3 shows the partial result of synchronizing two timelines. Following the notation established earlier (see, Fig.2), each label on the interval is associated with conditions, target metric values, stakeholders, policies, and durations. Combining these results in intervals which carry, at first simply combined labels, but later require checking for consistency—the principal indicator for checking and possible revision being the negative interactions (obtained from correlations) between metrics whose trade-offs are not accounted for in individual timelines, but do need to be treated in the synchronized timeline (i.e., a metric in $\mathcal{T}_{q_1}$ interacts negatively with a metric in $\mathcal{T}_{q_2}$). Observe in the middle of Fig.3 that the time is represented above the extended timeline forms so that the length of the intervals be proportional to their durations.

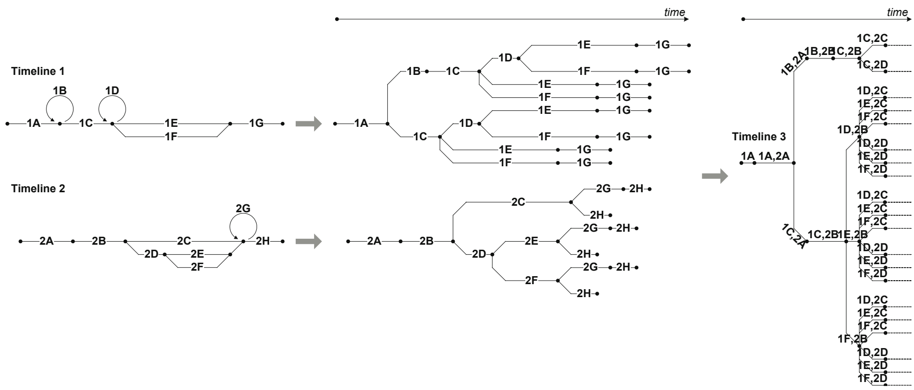



**Fig. 3.** Synchronizing two arbitrary timelines

**Technique 4.** *(GenerateAlternatives algorithm)* The algorithm takes two timelines $\mathcal{T}_{q_1}$ and $\mathcal{T}_{q_2}$ to produce a third timeline $\mathcal{T}_{q_3}$ which contains all allowed alternative paths obtained by combining $\mathcal{T}_{q_1}$ and $\mathcal{T}_{q_2}$. First, the set of timepoints is built for $\mathcal{T}_{q_3}$. Let

$$\mathcal{T}_{q_1} = (G_{q_1}, F_{q_1}),$$
$$\mathcal{T}_{q_2} = (G_{q_2}, F_{q_2}),$$
$$T_{q_1} = \{t_i \mid \forall n_i \in N_{q_1}, t_i = timing_{q_1}(n_i)\},\ and$$
$$T_{q_2} = \{t_i \mid \forall n_i \in N_{q_2}, t_i = timing_{q_2}(n_i)\}.$$

The first part of the algorithm generates the set of timed nodes $T_{q_3}$ for $\mathcal{T}_{q_3}$, provided a complete ordering of nodes in $\mathcal{T}_{q_1}$ and $\mathcal{T}_{q_2}$ is available (i.e., it is enough to associate arbitrary times to nodes, while only ensuring their order corresponds to what is expected by the stakeholders):

**begin GenerateAlternatives**
$i \leftarrow 0; \mathcal{T}_{q_3} \leftarrow \emptyset; N_{temp} \leftarrow N_{q_1} \cup N_{q_2}$
**while** $t_i \leq Max(T_{temp})$ and $t_i \in T_{temp}$ **do**
    $t_i \leftarrow Min(T_{temp})$
    **if** $t_i \neq t_{i-1}$ **then** $T_{q_3} \leftarrow T_{q_3} \cup \{t_i\}$ **end if**
    $T_{temp} \leftarrow T_{temp} \setminus \{t_i\}; i \leftarrow i+1$
**end while**

Paths from $\mathcal{T}_{q_1}$ and $\mathcal{T}_{q_2}$ are then combined to construct $P_{q_3}$ ($p^i_{q_3}$ denotes the i-th path in timeline $\mathcal{T}_{q_3}$). It is assumed that a function $getPaths : \{\mathcal{T}\} \longrightarrow \{P\}$ returns the set of all paths of a timeline:

$i \leftarrow 0; j \leftarrow 0; k \leftarrow 0; P_{q_1} \leftarrow getPaths(\mathcal{T}_{q_1}); P_{q_2} \leftarrow getPaths(\mathcal{T}_{q_2})$
**while** $k \leq |P_{q_1}|$ **do**
  **while** $j \leq |P_{q_2}|$ **do**
    **while** $t_i \leq Max(T_{q_3})$ **do**
      $p^i_{q_3} \leftarrow p^i_{q_3} \cup \{E_i : l(E_i) = Min(t_i \in p^k_{q_1}, t_i \in p^j_{q_2})$ and
        $r(E_i) = Min(\{t_i, t_{i+1} \in p^k_{q_1}\} \cup \{t_i, t_{i+1} \in p^j_{q_2}\} \setminus \{l(E_i)\})$

$E' \leftarrow E \in E_{q_1}, E \subseteq E'; E'' \leftarrow E \in E_{q_2}, E \subseteq E''$
$setMetric_{q_3}(E_i) \leftarrow (setMetric_{q_1}(E') \cup setMetric_{q_2}(E''))$
$agreedBy_{q_3}(E_i) \leftarrow (agreedBy_{q_1}(E') \cup agreedBy_{q_2}(E''))$
$condition_{q_3}(E_i) \leftarrow (condition_{q_1}(E') \cup condition_{q_2}(E''))$
$policy_{q_3}(E_i) \leftarrow (policy_{q_1}(E') \cup policy_{q_2}(E''))$
$P_{q_3} \leftarrow P_{q_3} \cup p^i_{q_3}; i \leftarrow i + 1$
**end while**
$j \leftarrow j + 1$
**end while**
$k \leftarrow k + 1$
**end while**

Let *pathsToEdges* be the function that takes the set of paths and returns the unique set of edges employed to build the given paths. The third timeline is then defined as follows:

$E_{q_3} \leftarrow pathsToEdges(P_{q_3})$
$F_{q_3} \leftarrow \{setMetric_{q_3}, agreedBy_{q_3}, condition_{q_3}, policy_{q_3}, timing_{q_3}\}$
$\mathcal{T}_{q_3} \leftarrow (T_{q_3}, E_{q_3}, F_{q_3})$
**end GenerateAlternatives**

*Example 4.* Fig.3 illustrates the use of the algorithm by visualizing the two input timelines and the result as a branching timeline, easily obtained from the formal timeline definitions.

By progressively combining all NFRs' timelines, the general NFR timeline can be constructed. As visible from the algorithm, visual check of the obtained timeline is required to ensure combined conditions, policies, and metric values are not inconsistent. If inconsistencies are detected (taking the form of, e.g., incompatible metric values or preferences, contradictory conditions or policies), they ought to be resolved by either adjusting the problematic paths or eliminating them. Ideally, these checks should be performed early, after each application of the algorithm so that problems are detected and resolved before moving further with timeline synchronization. It is apparent that combining timelines using the GenerateAlternatives algorithm results in complex trees difficult to analyze visually for more than a few NFRs. The following technique are used to reduce the set of alternative paths to consider.

**Technique 5.** *(Pruning criterion: Pareto efficiency)* Under the reasonable assumption of stakeholder rationality, and knowing that changing the metric target values may affect, if interaction is present, other metrics' target values, any path to maintain in the timeline must be Pareto efficient.[1] In the present terminology, a path is Pareto efficient if there is no interval on it such that a target metric value associated to this interval can be improved without degrading the target value of another metric on the same path. That degradation will occur is known from the correlation between metrics.

[1] In economics, an allocation of resources between economic agents is Pareto efficient if any change in the allocation intended to improve the part of some agents makes at least some other agents worse off.

Other criteria and associated techniques for path selection not discussed in the present paper, though relevant are:

- *Pruning by probable paths*: If statistical data is available on the ocurrence of conditions and the execution of processes fulfilling the policies indicated in the timeline (data may come from, e.g., comparable processes put in place in other project), probabilities can be associated with conditions and policies in the aim of selecting the most probable paths.
- *Pruning by risk*: Identifying least preferred paths may be of interest (this can be driven by identification of least-preferred intervals, which is simpler to execute than by observing entire paths), as the conditions associated to them clearly pose a risk in terms of quality levels. Knowledge of problematic conditions may provide indications useful for revising conditions and policies on preferred paths in order to make them more robust. Such revision can be executed by applying Defensive NFR planning over preferred paths.
- *Priority-based highest-metric pruning*: A classification of paths by aggregate priority level and aggregate metric target level may point automatically to prioritized and most preferred paths to consider.

As no single technique guarantees by itself an appropriate choice of path, combining several of the above techniques may provide insight from different perspectives and reduce the set of interesting paths to a manageable number.

## 4 Conclusions

This paper opens up a discussion of the specification and analysis of time properties of NFRs. The aim has been to suggest basic useful concepts and analyses and point to difficulties. More thorough study of both the specification and analysis of NFR temporal properties is necessary, as the variation of NFR satisfaction levels over time is a fact. Various temporal formalisms are available, and their relative worth should now be studied in light of the likely (un)availability of information. Methodological considerations will receive considerable attention, given the difficulty to predict and plan for the variation of NFR satisfaction levels. Our current focus is on additional analysis techniques, the integration of the concepts discussed herein with NFR, and the automation of the techniques.

## References

1. Eurocontrol. ATM User Requirements Document (Vol. 1 & Vol. 2). European Air Traffic Control Harmonisation and Integration Programme, Ref FCO.ET1.ST04.DEL01, European Org. for the Safety of Air Navigation (1999)
2. Mylopoulos, J., Chung, L., Nixon, B.: Representing and Using Nonfunctional Requirements: A Process-Oriented Approach. IEEE Trans. Softw. Eng. 18(6), 483–497 (1992)
3. Osterweil, L.: Strategic Directions in Software Quality. ACM Computing Surveys 28(4), 738–750 (1996)

# Preface to SeCoGIS 2008

Esteban Zimányi[1] and Christophe Claramunt[2]

[1] Université Libre de Bruxelles, Belgium
[2] Naval Academy Research Institute, France

Geographical Information Systems are nowadays considered as one of the most challenging and promising areas that might beneficiate from research advances in conceptual and semantic modeling. It has been largely recognised that the integration of semantic and conceptual models within Geographical Information Systems will help to develop innovative solutions for the benefits of urban and environmental applications. This is very much the objective of the SeCoGIS workshop that serves as an integrated forum for disseminating research and experience in conceptual and semantics issues associated to Geographical Information Systems. It brought together leading researchers and developers to discuss the state of the art, and to understand novel challenges and emerging research directions.

The SeCoGIS workshop has been originally initiated from two successful series of workshops, Conceptual Modeling for Geographic Information Systems (CoMoGIS), and Semantic-based Geographical Information Systems (SeBGIS). The SeCoGIS workshop is organized in a single-track specialised session to highly stimulate interaction amongst the participants. The co-location with the Entity-Relationship Conference allows cross-fertilization and mutual interactions between the ER and GIS research communities.

The call for papers attracted 20 papers, which were submitted by authors from 10 countries and 3 continents, clearly illustrating the international nature of the domain. The program committee consisting of 31 researchers conducted three to four reviews of each paper and selected 9 papers for presentation and discussion at the workshop.

The workshop was organised in three sessions of 3 papers each. The first session is devoted to foundational issues. The second session covers ontologies and location-based services. Finally, the third session addresses the issues of interoperability and spatial infrastructures.

Many people help in putting together this workshop. First of all the Steering Committee was in charge of the initial conception and reality of the workshop. The Program Committee has been very much efficient on carefully reviewing the papers under a very tight schedule. Special thanks to Philippe Rigaux for providing the MyReview program, which was used during the reviewing process.
We hope that you find the program and presentations beneficial and enjoyable and that you had during the workshop many opportunities to meet colleagues and practitioners.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, p. 312, 2008.

# Projective Relations on the Sphere

Eliseo Clementini

University of L'Aquila, Department of Electrical and Information Engineering,
67040 Poggio di Roio, L'Aquila, Italy
eliseo@ing.univaq.it

**Abstract.** For geographic applications, it is of great interest to study the geometric properties of the sphere. Spatial models that directly deal with geographic coordinates are able to contemplate such aspects that would be otherwise disregarded by flat 2D models. For example, on the Earth's surface, a point which is to the East of another point could be reached travelling to West as well. Being situated at the North Pole, taking any direction would mean travelling to South. In this paper, we study projective relations on the sphere, a category of spatial relations more general than cardinal directions and that, together with topological relations and qualitative distances, represent the essence of qualitative spatial relations.

**Keywords:** Projective relations, Qualitative spatial relations, Spherical geometry.

## 1 Introduction

Most Geographical Information Systems (GISs) deal with 2D spatial data. The interest of directly treating spherical coordinates has been recognized by many researchers, such as in [1, 2]. Developing geometric models of the globe is especially important in recent times with the increasing interest in global problems.

If quantitative spatial analysis tools for the Earth surface are not much developed, the study of qualitative relations for human-computer interfaces has never begun on the sphere, with the notably exception of the work on topological relations [3], extending the 9-intersection model. The research issue of spatial relations on the sphere was identified, among others, by Usery [4]: the passage from a flat model to a global model is not obvious especially on a small scale map, depicting entire countries, when the terrestrial curvature is not irrelevant.

In this paper, we describe a qualitative model for ternary projective relations among extended objects on the sphere. Models for projective relations among points in 2D [5] or among extended regions in 2D [6, 7] or 3D [8] are not directly applicable to a spherical surface. In detail, we take into consideration the 5-intersection model for 2D extended regions [6], revising the definitions of the basic relations and extending it with some new relations that can only be defined on the sphere. We obtain a set of jointly exhaustive and pair-wise disjoint (JEPD) set of 42 projective relations on the sphere (against the 34 on the plane). Finally, we apply the model to describe cardinal directions on the Earth, because cardinal directions can be considered a special case of projective relations.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 313–322, 2008.

In Section 2, we describe preliminary concepts about the geometry of the sphere. In Section 3, we briefly recall the 5-intersection model in 2D. In Sections 4 and 5, we introduce the model for projective relations among points and regions on the sphere, respectively. In Section 6, we customize the model of Section 5 to deal with cardinal directions on a spherical Earth. In Section 7, we draw short conclusions.

## 2 The Geometry of the Sphere

As a general introduction to the subject, in [9] authors describe eleven properties of the sphere (i.e., the 2D spherical surface) and discuss whether these properties uniquely determine the sphere. On the sphere, there are no straight lines in the usual sense. Straight lines in the plane or in the space are characterized by the fact that they are the shortest paths between points. The curves on the sphere with the same property are the *great circles*. A great circle subdivides the sphere in two hemispheres. Taking great circles as a replacement for lines, it is possible to define polygons and other geometric objects on the sphere. A notable property of great circles is that, given two distinct points $A$ and $B$ on the sphere, if the points are not antipodal, they determine a unique great circle, while if they are antipodal there are infinitely many great circles containing $A$ and $B$. Such a property is called the first incidence relation for the sphere. The second incidence relation is that two distinct great circles meet in exactly two antipodal points.

Contrary to the plane, on the sphere there exist polygons with two sides: any two great circles meeting in two antipodal points divide the sphere into four regions, each of which has two sides. Such a region is called a *lune* or a biangle, from the analogy to the part of the moon that is illuminated by the sun and at the same time visible from the Earth. The vertices of a lune are antipodal points and the two angles of a lune are equal. A definition of a region on the sphere is not as intuitive as a region on the plane. In fact, a region on the plane can be defined as the portion that is inside a closed curve. The same definition on the sphere would be ambiguous because it is not straightforward what part of the sphere should be considered the "inside" and what part the "outside", as any closed curve on the sphere subdivides the sphere in two finite parts. Therefore, to avoid the ambiguity in defining a region, it is necessary to state explicitly which part of the sphere is the interior of the region. Similarly to definitions in 2D [10], regions are *simple* if they are regularly closed (i.e., $A = \overline{A^\circ}$) and without holes and disconnected components. Regions are *complex* if they are regularly closed and have holes and disconnected components. The model that is developed in the paper refers to complex regions without restrictions.

The convex hull of a region $A$, indicated as $CH(A)$, is the intersection of all the hemispheres that contain $A$. The convex hull of a region can be defined if the region is entirely contained inside a hemisphere. A convex region is always contained inside a hemisphere. If a region cannot be contained inside a hemisphere, its convex hull degenerates to the whole sphere.

## 3 The 5-Intersection Model on the Plane

A ternary relation among three objects *A*, *B*, and *C* is denoted with *r*(*A*,*B*,*C*). We assume that the first object *A* involved in the relation *r* has the role of primary object and the second and third objects have the role of reference objects. Therefore the relation *r*(*A*,*B*,*C*) should be read as "*A* is in relation *r* with *B* and *C*". The order of the reference objects *B* and *C* affects the orientation, that is, an orientation from reference object *B* to reference object *C* is assumed.

The basic projective relation in the plane is the *collinearity* of three points. Three points *x*,*y*,*z* are *collinear* if they lie on the same line. Given an oriented line connecting the two reference points *y* and *z*, the plane is subdivided in two half-planes and the line itself is subdivided in three parts. This gives rise to the five projective relations *between* (*bt*), *rightside* (*rs*), *leftside* (*ls*), *before* (*bf*), *after* (*af*). If two of the points are coincident, the three points are trivially collinear, since two points determine a line. If all three points are coincident, they are also trivially collinear because there are infinite lines incident in a point. In the case the two reference points are coincident, we refine the relation *collinear* in the relations *inside* (*in*) and *outside* (*ou*). The set of ternary relations among points *rs*, *ls*, *bt*, *bf*, *af*, *in*, *ou* is a JEPD set of relations, as it was proved in [6].

The definitions for points are a special case of the definitions for regions. The relation *collinear* among regions can be introduced as a generalization of the same relation among points. For a complete discussion about definitions of collinearity for regions see [11]. If two reference regions *B* and *C* have disjoint convex hulls, then two pairs of common tangents are uniquely defined: the external common tangents and the internal common tangents (Fig. 1(a)). The internal common tangents subdivide the plane in four cones. The external common tangents define the convex hull of the union of the two reference regions: such a convex hull corresponds to the acceptance area of the relation *bt*. By intersecting the convex hull with the four cones, we obtain the acceptance areas of the relations *rs*, *ls*, *bf*, *af*. As a special case, if the two reference regions have not disjoint convex hulls, internal common tangents cannot be defined and the plane can be partitioned in two areas, corresponding to relations *in* and *ou* (Fig. 1(b)).

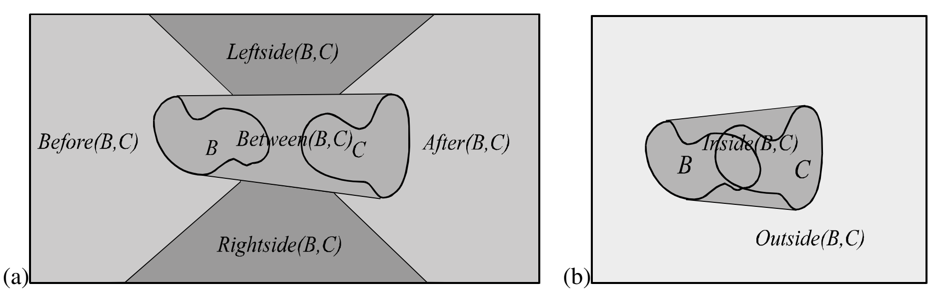



**Fig. 1.** (a) The partition of the plane into five areas for disjoint convex hulls of reference regions; (b) the partition into two areas for non disjoint convex hulls of reference regions

The *5-intersection* matrix can be defined by considering the empty/non-empty intersections of the primary region $A$ with the five acceptance areas:

| | | |
|---|---|---|
| | $A \cap Leftside(B,C)$ | |
| $A \cap Before(B,C)$ | $A \cap Between(B,C)$ | $A \cap After(B,C)$ |
| | $A \cap Rightside(B,C)$ | |

The 5-intersection can have $2^5–1$ different configurations. Each configuration corresponds to a projective relation among three regions $A$, $B$, and $C$, where $CH(B) \cap CH(C) = \varnothing$. The matrix with all five values equal to zero does not correspond to a relation. For $CH(B) \cap CH(C) \neq \varnothing$, we consider the following *2-intersection* matrix:

| | |
|---|---|
| $A \cap Inside(B,C)$ | $A \cap Outside(B,C)$ |

which can assume the values (0 1), (1 0), and (1 1). Overall, the model identifies a JEPD set of 34 projective relations among three regions of the plane.

## 4 Projective Relations among Points on the Sphere

The most primitive projective relation is collinearity, as discussed in [11]. Equivalently, on the sphere three points are said to be *collinear* if they lie on the same great circle. If they do not lie on the same great circle, they are said to be *aside*. Notice that three points can be aside if none of them makes a pair of antipodal points nor coincident points; otherwise they are collinear. Let us find an equivalent of the 5-intersection model for three points of the sphere. The line passing through two distinct reference points $y$ and $z$ should be substituted by the great circle passing through $y$ and $z$, which is unique if $y$ and $z$ are not antipodal points. However, the concept of direction from point $y$ to point $z$ should be better defined: if on the plane this concept is unique, on the sphere we can connect $y$ to $z$ following the two opposite directions along the great circle. Therefore, it seems opportune to define the direction from $y$ to $z$ as the one following the shortest path. Equivalently, the chosen direction is the direction that permits to reach $z$ from $y$ remaining in the same hemisphere as $y$ and $z$. We indicate with $[y,z]$ the arc from $y$ to $z$, extremes included. We indicate the longer arc from $z$ to $y$ with $(z,y)$, extremes excluded. In this way, we are able to define a partition of the sphere in 4 parts: the arc $[y,z]$, the remaining part of the great circle $(z,y)$, and the two hemispheres that are determined by the great circle (see Fig. 2). The hemisphere to the right of the arc $[y,z]$ will be indicated as $HS^{+}_{yz}$ and the hemisphere to the left will be indicated as $HS^{-}_{yz}$. The two hemispheres correspond to the relations *rs* and *ls*. The arc $[y,z]$ corresponds to the relation *bt*. The arc $(z,y)$ corresponds to the relation *nonbetween* (*nonbt*). With respect to the 5-intersection for points on the plane, on the sphere we cannot subdivide the relation *nonbetween* in the two relations *af* and *bf*. To build a JEPD set of relations, we additionally introduce the

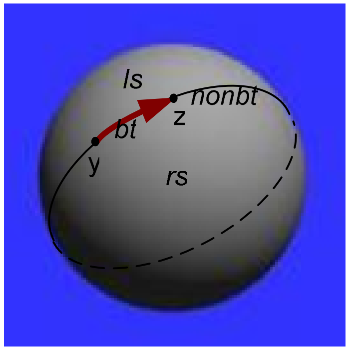



**Fig. 2.** Projective relations identified by two reference points

**Table 1.** The definitions of projective relations among points on the sphere

| name | short name | definition |
|---|---|---|
| *collinear* | *coll*(*x*,*y*,*z*) | $\exists\ great\ circle\ c : x \in c,\ y \in c,\ z \in c$ |
| *aside* | *aside*(*x*,*y*,*z*) | $\neg\ coll(x,y,z)$ |
| *rightside* | *rs*(*x*,*y*,*z*) | $x \in HS^{+}_{yz}$ |
| *leftside* | *ls*(*x*,*y*,*z*) | $x \in HS^{-}_{yz}$ |
| *inside* | *in*(*x*,*y*,*z*) | $x=y\ \wedge\ y=z$ |
| *outside* | *ou*(*x*,*y*,*z*) | $x \neq y\ \wedge\ y=z$ |
| *in_antipodal* | *in_ant*(*x*,*y*,*z*) | $(x=y\ \vee\ x=z)\ \wedge\ antipodal(y,z)$ |
| *out_antipodal* | *out_ant*(*x*,*y*,*z*) | $x \neq y\ \wedge\ x \neq z\ \wedge\ antipodal(y,z)$ |
| *between* | *bt*(*x*,*y*,*z*) | $y \neq z\ \wedge\ \neg\, antipodal(y,z)\ \wedge\ x \in [y, z]$ |
| *nonbetween* | *nonbt*(*x*,*y*,*z*) | $y \neq z\ \wedge\ \neg\, antipodal(y,z)\ \wedge\ x \in (z, y)$ |

relations *inside*, *outside*, *in_antipodal*, *out_antipodal*, which take care of particular cases (see Table 1).

## 5 Projective Relations among Regions on the Sphere

Two regions *B* and *C* are contained in the same hemisphere, *same_hemisphere*(*B*,*C*), if it is not possible to find a pair of antipodal points *y* and *z* in *CH*(*B*) and *CH*(*C*), respectively:

$$same\_hemisphere(B,C) = \forall y \in CH(B)[\forall z \in CH(C)[\neg antipodal(y,z)]]$$

For a region that contains two antipodal points, it is not possible to build the convex hull, in the sense that the convex hull coincides with the sphere.

When applied to the sphere, the 5-intersection model for regions maintains the desirable property of subdividing the space in five distinct acceptance areas and identifying five basic projective relations (single-tile) and their combinations (multi-tile relations).

This allows us to continue to use the same theoretical developments that applied to the plane in the case of the sphere.

Given two reference regions *B* and *C*, embedded on the sphere, in making the construction of the five acceptance areas in which the sphere is subdivided, the first step is to consider the equivalent of internal and external tangents to regions *B* and *C*. These tangents are obtained by replacing tangent lines with tangent great circles. A tangent great circle to a region (say *B*) is defined as follows: if we take the plane containing the great circle and the tangent plane to the sphere at the point of intersection between the great circle and region *B*, the intersection between these two planes is a line tangent to region *B*. We can define an internal tangent for regions *B* and *C* on the sphere as a great circle that is tangent to both regions *B* and *C* and leaves the two regions in two opposite hemispheres. We can define an external tangent for regions *B* and *C* on the sphere as a great circle that is tangent to both regions *B* and *C* and leaves the two regions in the same hemisphere.

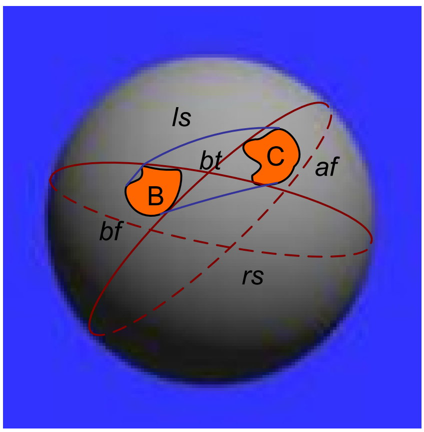



**Fig. 3.** Projective relations identified by two reference regions

Obviously, it is possible to make the above construction when internal and external tangents for the two reference regions do exist. The two internal tangents subdivide the sphere in four lunes: reference regions must be contained in a pair of opposite lunes (see Fig. 3). The condition for the existence of internal tangents is that the convex hulls of the reference regions must be disconnected. Regarding external tangents, they leave the two reference regions in the same hemisphere. It is not possible to find external tangents if reference regions *B* and *C* contain two antipodal points. Therefore, the two reference regions must be contained in the same hemisphere. Joining the two conditions, to apply the 5-intersection model, the two reference regions must be contained in the same hemisphere and their convex hulls must be disconnected. The convex hull of the union of *B* and *C* determines the *Between*(*B*,*C*) acceptance area. The four lunes that are determined by the internal tangents minus the *Between*(*B*,*C*) area define the *Before*(*B*,*C*), *After*(*B*,*C*), *Rightside*(*B*,*C*), and *Leftside*(*B*,*C*) acceptance areas. The *Before*(*B*,*C*) and *After*(*B*,*C*) areas touch in the antipodal point of the internal tangents' intersection. The same happens for the *Rightside*(*B*,*C*) and *Leftside*(*B*,*C*) areas.

**Table 2.** The definitions of projective relations among regions in the sphere

| relation | definition | acceptance area |
|---|---|---|
| *coll*(*A*,*B*,*C*) | $\forall x \in A^\circ\ [\exists y \in CH(B)^\circ\ [\exists z \in CH(C)^\circ\ [coll(x,y,z)]]]$ | $Coll(B,C) = Lune_{bf}(B,C) \cup Lune_{af}(B,C) \cup (CH(B \cup C))^\circ$ |
| *aside*(*A*,*B*,*C*) | $\forall x \in A^\circ\ [\forall y \in CH(B)^\circ\ [\forall z \in CH(C)^\circ\ [\neg\, coll(x,y,z)]]]$ | $Aside(B,C) = Lune_{rs}(B,C) \cup Lune_{ls}(B,C) - (CH(B \cup C))$ |
| *rs*(*A*,*B*,*C*) | $\forall x \in A^\circ\ [\forall y \in CH(B)^\circ\ [\forall z \in CH(C)^\circ\ [rs(x,y,z)]]]$ | $Rightside(B,C) = Lune_{rs}(B,C) - CH(B \cup C)$ |
| *ls*(*A*,*B*,*C*) | $\forall x \in A^\circ\ [\forall y \in CH(B)^\circ\ [\forall z \in CH(C)^\circ\ [ls(x,y,z)]]]$ | $Leftside(B,C) = Lune_{ls}(B,C) - CH(B \cup C)$ |
| *in*(*A*,*B*,*C*) | $same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) \neq \varnothing \wedge A^\circ \subseteq (CH(B \cup C))^\circ$ | $Inside(B,C) = (CH(B \cup C))^\circ$ |
| *ou*(*A*,*B*,*C*) | $same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) \neq \varnothing \wedge A^\circ \cap (CH(B \cup C)) = \varnothing$ | $Outside(B,C) = S^2 - (CH(B \cup C))$ |
| *bt*(*A*,*B*,*C*) | $same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) = \varnothing \wedge A^\circ \subseteq (CH(B \cup C))^\circ$ | $Between(B,C) = (CH(B \cup C))^\circ$ |
| *nonbt*(*A*,*B*,*C*) | $same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) = \varnothing \wedge A^\circ \subset (Lune_{bf}(B,C) \cup Lune_{af}(B,C)) - CH(B \cup C)$ | $NonBetween(B,C) = (Lune_{bf}(B,C) \cup Lune_{af}(B,C)) - CH(B \cup C)$ |
| *bf*(*A*,*B*,*C*) | $same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) = \varnothing \wedge A^\circ \subset Lune_{bf}(B,C) - CH(B \cup C)$ | $Before(B,C) = Lune_{bf}(B,C) - CH(B \cup C)$ |
| *af*(*A*,*B*,*C*) | $same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) = \varnothing \wedge A^\circ \subset Lune_{af}(B,C) - CH(B \cup C)$ | $After(B,C) = Lune_{af}(B,C) - CH(B \cup C)$ |
| *B_side*(*A*,*B*,*C*) | $\neg\, same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) = \varnothing \wedge A^\circ \subset Lune_B(B,C)$ | $B\_side(B,C) = Lune_B(B,C)$ |
| *C_side*(*A*,*B*,*C*) | $\neg\, same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) = \varnothing \wedge A^\circ \subset Lune_C(B,C)$ | $C\_side(B,C) = Lune_C(B,C)$ |
| *BC_opposite*(*A*,*B*,*C*) | $\neg\, same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) = \varnothing \wedge A^\circ \subset (Lune_{opp1}(B,C) \cup Lune_{opp2}(B,C))$ | $BC\_opposite(B,C) = Lune_{opp1}(B,C) \cup Lune_{opp2}(B,C)$ |
| *entwined*(*A*,*B*,*C*) | $\neg\, same\_hemisphere(B,C) \wedge CH(B) \cap CH(C) \neq \varnothing$ | $Entwined(B,C) = S^2$ |

To make a JEPD set of relations, we have to contemplate various special cases. If the two regions lie on the same hemisphere, but their convex hulls are not disconnected, it is possible to build the convex hull of the union of the two regions: a primary region *A* can be either *inside* or *outside* this convex hull.

If the two regions do not lie on the same hemisphere, but they lie on two opposite cones, there are no external tangents. Hence, it is not possible to define a between area. Also, it is not possible to define the shortest direction from *B* to *C*. Still the sphere is subdivided in four lunes by the internal tangents. A primary region *A* can be inside the lune of *B* (relation *B_side*), inside the lune of *C* (relation *C_side*), or inside one of the two other lunes (relation *BC_opposite*).

If the two regions' convex hulls are not disjoint and they do not lie on the same hemisphere, there are no internal tangents and the convex hull of their union coincides with the sphere. Therefore, this case can be considered as a separate projective relation, which we call *entwined*. In Table 2, there is a summary of definitions for all the relations discussed so far. The JEPD set of projective relations for three regions on the sphere is given by all possible combinations of the following basic sets:

- *between*, *rightside*, *before*, *leftside*, *after* (31 combined relations);
- *inside*, *outside* (3 combined relations);
- *B_side*, *C_side*, *BC_opposite* (7 combined relations);
- *entwined* (1 relation).

In summary, in the passage from the plane to the sphere, we identify 8 new basic relations. The set of JEPD relations is made up of 42 relations.

## 6 Mapping the 5-Intersection to Cardinal Directions

Cardinal directions have been studied as a particular category of qualitative spatial relations between 2D points [12] and regions [7]. Cardinal directions between extended regions over geographic coordinates of the Earth can be easily expressed as a particular case of the 5-intersection model on the sphere. We make the simplified assumption that the Earth is spherical. We assume a set of four cardinal directions (*North*, *East*, *South*, *West*) applied between two regions: a reference region *R*2 and a primary region *R*1. Let us make a correspondence between a cardinal direction and a projective relation on the sphere as follows: replace the reference region *C* with the North pole, the reference region *B* by the reference region *R*2 of the cardinal direction, and the primary region *A* by the primary region *R*1 of the cardinal direction. The acceptance area for *North* is the *Between*(*R*2, North Pole). The acceptance areas for *South*, *East*, and *West* are the *Before*, *Rightside*, and *Leftside*, respectively. The *After* area corresponds to the antipodal lune of that containing *R*2 and we assume that it is undetermined with respect to the cardinal direction. The two internal tangents are the two meridians touching regions *R*2. For example, in Fig. 4, region *R*2 is Europe: Africa is south of Europe, Americas are west of Europe, Asia is east of Europe.

The *North* region, as we just defined it, corresponds to the *Between* region, including therefore the reference region. An alternative definition for the *North* is the difference between the *Between* area and the convex hull of the region *R*2. This would

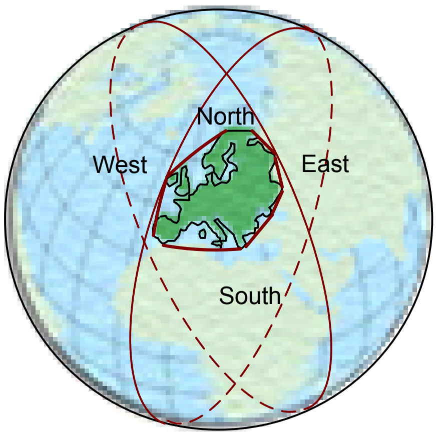



**Fig. 4.** Mapping of the four cardinal directions to the 5-intersection

also make a symmetry between *North* and *South* regions, which would both be defined as the two parts in which the lune containing the reference region *R*2 is separated by *R*2 itself.

## 7 Conclusions

In this paper, we illustrated a model for projective relations on a spherical surface. We showed that the most of the concepts that were developed for projective relations on the plane can be reused by some adaptations. It is interesting that the model for points loses one relation passing from the plane to the sphere, since no after/before distinction can be made. Instead, the 5-intersection for regions in the sphere is able again to make such a distinction. To obtain a JEPD set of relations, we had to consider various peculiar cases that did not exist on the plane and, therefore, new relations on the sphere. Further work would be needed to find the composition table for reasoning with projective relations on the sphere and to integrate them with topological relations and qualitative distances. Finally, we showed an application of the model to the four cardinal directions in geographic coordinates, thus providing sound definitions of cardinal directions among extended regions on the sphere. With regard to this application, further work is necessary to refine the basic geometric categorization in four directions, taking also into account user and context-dependent aspects that influence the way people reason with cardinal directions. The diffusion of web tools, such as Google Earth, also suggests to investigate how to integrate qualitative projective relations within them.

## References

1. Tobler, W.: Global spatial analysis. Computers, Environment and Urban Systems 26, 493–500 (2002)
2. Dutton, G.: A hierarchical coordinate system for geoprocessing and cartography. Lecture Notes in Earth Science, vol. 79. Springer, Berlin (1999)

3. Egenhofer, M.J.: Spherical Topological Relations. In: Spaccapietra, S., Zimányi, E. (eds.) Journal on Data Semantics III. LNCS, vol. 3534, pp. 25–49. Springer, Heidelberg (2005)
4. Usery, L.: Global Representation and Modeling. University Consortium for Geographic Information Science Research Priorities (2002)
5. Freksa, C.: Using Orientation Information for Qualitative Spatial Reasoning. In: Frank, A.U., Campari, I., Formentini, U. (eds.) GIS 1992. LNCS, vol. 639, pp. 162–178. Springer, Heidelberg (1992)
6. Clementini, E., Billen, R.: Modeling and computing ternary projective relations between regions. IEEE Transactions on Knowledge and Data Engineering 18(6), 799–814 (2006)
7. Goyal, R., Egenhofer, M.J.: The Direction-Relation Matrix: A Representation of Direction Relations for Extended Spatial Objects. In: UCGIS Annual Assembly and Summer Retreat, Bar Harbor, ME (1997)
8. Billen, R., Clementini, E.: Projective relations in a 3D environment. In: Raubal, M., Miller, H.J., Frank, A.U., Goodchild, M.F. (eds.) GIScience 2006. LNCS, vol. 4197, pp. 18–32. Springer, Heidelberg (2006)
9. Hilbert, D.: Geometry and the Imagination, 2nd edn. AMS Chelsea Pub., Providence (1999)
10. Clementini, E., Di Felice, P.: A Model for Representing Topological Relationships Between Complex Geometric Features in Spatial Databases. Information Sciences 90(1-4), 121–136 (1996)
11. Billen, R., Clementini, E.: Semantics of collinearity among regions. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM-WS 2005. LNCS, vol. 3762, pp. 1066–1076. Springer, Heidelberg (2005)
12. Frank, A.U.: Qualitative Reasoning about Distances and Directions in Geographic Space. Journal of Visual Languages and Computing 3(4), 343–371 (1992)

# Life and Motion Configurations: A Basis for Spatio-temporal Generalized Reasoning Model

Pierre Hallot and Roland Billen

Geomatics Unit, University of Liege, 17 Allée du 6-Août,
B-4000 Liege, Belgium
{P.Hallot,rbillen}@ulg.ac.be

**Abstract.** Although intensive work has been devoted to spatio-temporal qualitative reasoning models, some issues such as management of complex objects life and motion remain. In this paper, we propose a model dealing with *existence* and *presence* of object concepts. First, we introduce *spatio-temporal states*, which express *existing spatio-temporal relationships between two objects at a given time*. Spatio-temporal states decision tree is presented. Based on this new representation, we construct a finite set of *life and motion configurations* which can be seen as a way to categorise *spatio-temporal histories.* Then, we present the model itself which is based on *25 generalized life and motion configurations.* Indeed, these generalized configurations are assimilated to *line-line topological relationships* obtained by projecting life and motion configurations in a *primitive space*. Finally, generalized life and motions configurations conceptual neighbourhood diagram and their interpretation in natural language are given.

**Keywords:** Spatio-temporal reasoning, spatio-temporal states, life and motion configuration, primitive space, spatio-temporal generalization, natural language interpretation.

## 1 Introduction

For years now, spatio-temporality has been key research issue in Geographic Information Science [1]. Indeed, temporality is closely related to spatiality for most geographical applications. Furthermore huge amount of spatio-temporal data is now available; spatio-temporal acquisition techniques such as the GNSS, RFID, Wi-Fi…are becoming casual. Researches in this area concern data acquisition techniques, information modelling, information management, visualization, analysis, ontologies, continuity [2]…. Potential applications are numerous; movement description, crime mapping, epidemiology, behaviour monitoring, robot navigation, way finding, battlefield information analysis system, logistics, fleet management, cultural heritage management... It is therefore crucial to develop efficient spatio-temporal analyses and reasoning processes to fully exploit these incredible information sources. Spatio-temporal modelling concerns spatio-temporal objects and also spatio-temporal relationships between these objects. Description of spatial relationships, spatial representation and spatial reasoning are typical *qualitative spatial reasoning* topics [3]. This branch of AI has provided comprehensive

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 323–333, 2008.

spatial calculus such as RCC [4] and 9-intersection [5]. Integrating time in such models was a quite logical evolution [6, 7]. For instance, Gerevini and Nebel [8] have "temporalized" RCC with the temporal logic of Allen. They have proposed the Spatio-temporal Constraint Calculus (STCC) which is based on a combination of the 8 connection relationships (RCC-8) between two regions [9] and the 13 temporal intervals relationships proposed by Allen [10] and which describes spatial constraints during a valid time interval. Similarly, Wolter and Zakharyaschev [7, 11] have "temporalized" the RCC with the propositional temporal logic (PTL). There exists another way to develop spatio-temporal reasoning models. As mentioned in [6, 12], spatio-temporal reasoning is not just the addition of space with time; the world itself can be seen as composed of spatio-temporal entities. Thus, it is sensible to develop spatio-temporal reasoning based on spatio-temporal shapes [13]. Muller has proposed an integrated vision of spatio-temporal entities which is described in [14]. He considers space-time histories of objects as primitive entities and analyses directly spatio-temporal shapes or histories. He defines a specific space-time to characterise classes of spatial changes, which is the first full mereotopological theory based on space-time as a primitive. Recent models also use integrated qualitative description of space and time parameters such as distance, speed, acceleration to reason with and infer new knowledge. The analysed values are represented qualitatively in regard of their stability or not. For example, the qualitative trajectory calculus (QTC) [15] in its basic form describes in a qualitative way movement between two moving object assimilated to points. It assumes that two moving objects are disjoint most of the time. It is worth mentioning a work realized by Noyon et al [16] which introduces a relative representation of trajectories in space and time. Their objective was to represent space as it is perceived by a moving observer using basic primitives such as relative position and velocities. However, some spatio-temporal issues are not fully handled by these models and still have to be explored. We wish to point out three of them. First, most of trajectories descriptions models focus on collision detection and postulate that two moving objects share most of the time spatial disjoint relationships during their evolution. In many cases, it would be interesting to analyse what's happening when two objects share the same place during a certain period of time, e.g. in the field of crime mapping when a murderer meets his victim or in epidemiology when a carrier meet a healthy person during more than on instant Secondly, most of the models consider only coexisting objects. However, it could happen that an object "disappeared" for a while (e.g. an object leaves the analysed zone or information's cut occurs such as a GPS cycle-slip). Finally, using these reasoning models requires implementing new spatio-temporal operators (in STDMS or STIS). The model proposed in this paper attempts to answer these issues. Our aim is to develop a spatio-temporal reasoning model valid for coexisting and non coexisting moving objects, considering new vision of spatio-temporal relationships between them. At this stage of our research, we assimilate objects to points. This new representation of spatiality including life and existence properties is used to create a complete set of life and motion configurations. Theses spatio-temporal configurations represent all the possible interactions between two points from topological and temporal points of view. Having identified possible life and motion configurations, we wish to propose a spatio-temporal qualitative calculus that uses existing topological operators. The selected operators are those of the 9-intersection established for line-line relations [5]. To apply such operators, we need to link spatio-temporal histories to life and motion configuration and project them in a primitive space

[14], i.e. a space where spatial and temporal dimensions are not differentiated. This projection allows us to generalize life and motion configurations into a small finite set of primitive relations between two moving objects (corresponding to two lines) with enough remaining information to perform spatio-temporal analyses. Thus, in a preliminary step, we will be able to reason about spatio-temporal information using already implemented topological operators. The paper is structured as follow. First we present the concept of "spatio-temporal states" and their representation. Then, we construct life and motion configurations by combining theses "spatio-temporal states". After that, we present our reasoning model which is the projection of the life and motion configurations in a primitive space and some common sense explanation in natural language of the generalised spatio-temporal relationships. Finally, we conclude.

## 2 "States" Model and Spatio-temporal Representation

### 2.1 Spatio-temporal States

Spatio-temporal evolution of objects can be rather complex; it is not limited to sharing or not sharing common place during a given time interval. Questions like existence, appearance, presence… occur: does a baby exist before his birth? Will this retired soccer player play again? Does this guy who is not working in a company exist for this company? Does a key is still present when in a pocket? In the preliminary step of our research, we have decided to reduce somehow the scope of spatio-temporal evolution by considering only moving points. We do not deal with other spatial representations such as regions or solids, and consequently we do not consider change of shapes. In terms of temporality, we wish to consider first continuous concepts such as existence and presence. By continuous we mean concept that last during a certain time interval. Then we will discuss instantaneous concepts, such as birth or death, appearance and disappearance. We are going to define different kinds of spatio-temporal states between two objects. By state we mean an *existing relationship between two objects at a given time*. We start with concept of *existence*. There is a time period where an object has not yet existed and one where it will not exist anymore. We also believe that an object which stopped to exist could not revive and that objects must have existed at least more than one instant to be considered. At a given time, four possibilities occur between points A and B: A does not exist and B does not exist $\{\nexists A \wedge \nexists B\}$, A exist and B does not exist $\{\exists A \wedge \nexists B\}$, A does not exist and B exist $\{\nexists A \wedge \exists B\}$ and finally, A and B exist $\{\exists A \wedge \exists B\}$. When existent, an object can be *present* or not. Examples of existing but non present objects could be a pedestrian who left a studied street, a thief who is in jail… In all theses cases, the object continue to exist but is not in the analysed space or it is not visible in this space. At a given time, four possibilities occur between points A and B: A is not present and B is not present $\{(A) \wedge (B)\}$, A is present and B is not present $\{A \wedge (B)\}$, A is not present and B is present $\{(A) \wedge B\}$, A is present and B is present $\{A \wedge B\}$. Note that the presence concept depends on the existence concept. It is crucial to consider these existence and presence concepts to fully encompass spatio-temporal complexity. Up to now, we have considered temporality only. When two objects are present (and therefore are existent), it becomes possible to consider spatial relationships between them at a given time. Topological relationships are by definition [5] associated to two spatial objects; a topological relationship can be identified between two objects if and only if they exist and are present. Possible topological relationships between points are “equal” or “disjoint”. In other words, at a given time, two possibilities occur between points A and B: A and B are equal {e} or A and B are disjoint {d}. Combining all these different possibilities regarding dependence relations between concepts, we obtain ten possible states between two objects and a decision tree (Fig. 1). The “*states*” compose a jointly exhaustive and pairwise disjoint set (JEPD).

The conceptual neighbourhood diagram of spatio-temporal states has been sketched, the dominance theory of Galton [17] has also been applied.

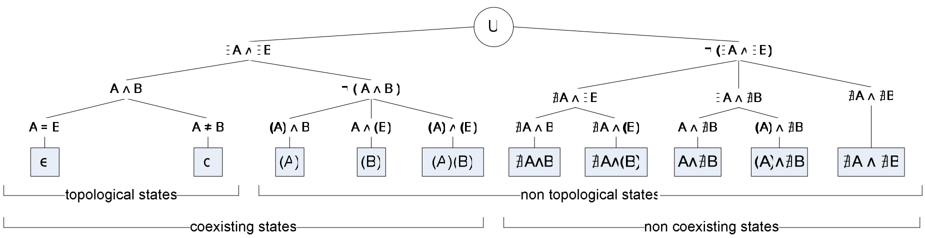



**Fig. 1.** Decision tree of the ten possible “states”. (A) means that the object A is not present. States can be gathered into topological states and non topological states or coexisting states and non coexisting states.

## 3 Life and Motion Configurations

Let us consider object’s *spatio-temporal history*. According to Hayes [18], (spatio-temporal) regions traced over time are termed spatio-temporal histories. In a n-D space, the spatio-temporal history is a n+1 dimensional volume. Remembering that we consider only points in this study, spatio-temporal history of an object is a line, continuous or not depending on object’s presence. Note that line’s extremities are not necessarily their birth or death as an object can exist without being present. The following figures represent a representation of spatio-temporal histories in the framework of spatio-temporal space [13].

The figure 2A corresponds to a moving points in a 1D space (trains on railroads) with this interpretation in temporal space and the figure 2B represents two moving points in a 2D space (men moving at the surface of the Earth). In these cases, the starting point corresponds to the birth and the ending point corresponds to the death of the object. There is infinity of life and motion configurations when considering spatio-temporal histories in a spatio-temporal space (one axis being time and the others being related to the Euclidean space dimension where points move). However, it is possible to characterise spatio-temporal histories using spatio-temporal states as defined in the previous section. Indeed, spatio-temporal histories can be summarised into a set of successive states. Possible successions of states will define a finite number of spatial configurations. We have established the entire set of possible spatio-temporal configurations regarding spatio-temporal states. At this step of the

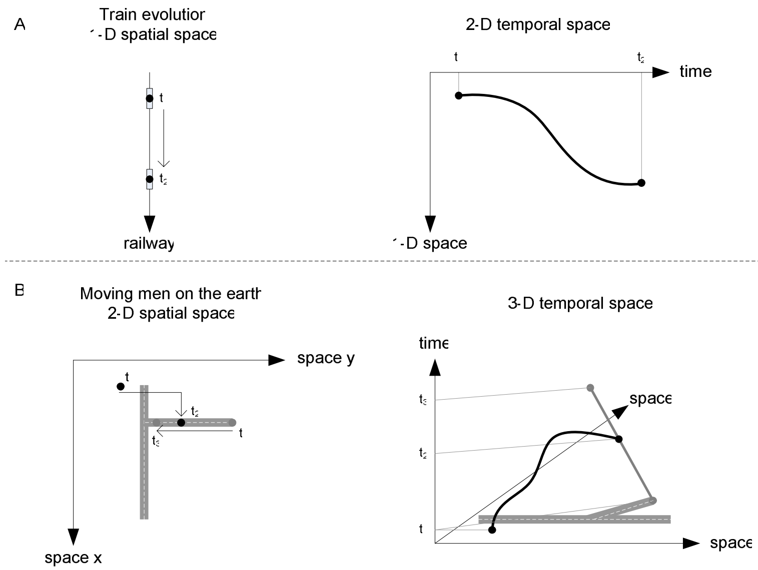



**Fig. 2.** Representation of spatio-temporal situations with the concept of temporal space. Case A presents moving train on a railroad and its interpretation in 2-D temporal space. Case B presents two moving persons on roads and their representation in 3-D temporal space. First dies when meeting the second.

research, we will present only the ones related states where objects have a continuous presence. A good life and motion configurations representation of spatio-temporal histories can be done adopting a degenerated notion of spatio-temporal space where the space axis is limited to the representation of three positions to have efficient visual expression of topological relationships disjunction and equality. Formal construction of life and motion configurations relying on states succession is presented in [19].

Spatio-temporal histories of points in a n-dimensional space can be fully described with a life and motion configuration represented in a 2-dimentional space, because their respective position in space, whatever space dimension, is reduced to two possible topological relationships.

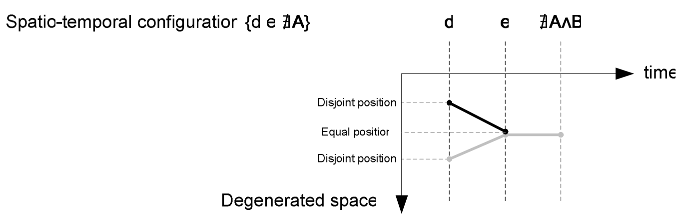



**Fig. 3.** Representation of life and motion configuration {d,e, ∄A }represented in degenerated temporal space. Spatial dimension is limited to three possible positions for representing relationships (disjoint or equal).

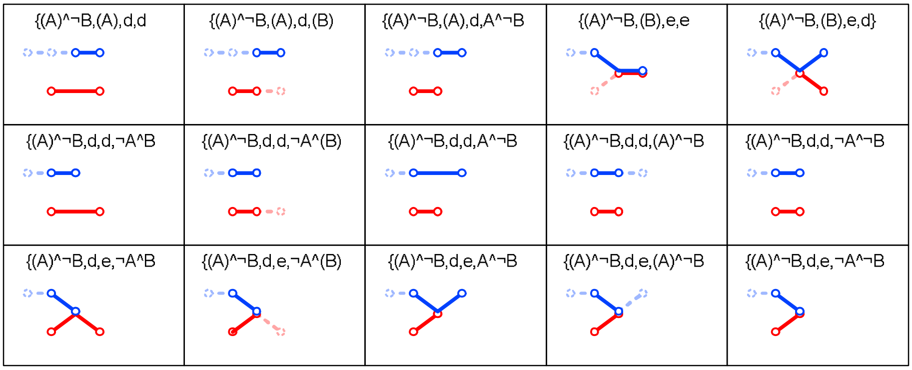



**Fig. 4.** Extract of the life and motion configuration of level 3. The spatial axis of the temporal space is vertical and the temporal one horizontal.

Life and motion configurations are obtained by ordering the different states in time (fig. 3). The set ε of states is composed by ($e_1$, $e_2$, …, $e_{10}$), the ten possible state values. They are combined a finite number of time n which is what we call the level of the life and motion configuration. Life and motion configurations are denoted as $\{e_1,e_2,\ldots,e_n\}$. It would not be possible to represent here the entire set of created spatio-temporal configuration. Some examples are represented in figure 4. In the following, we consider life and motion configurations created under two properties: objects could not revive and objects have a continuous presence, i.e. the spatio-temporal history is a continuous line. Based on these properties, there exist 4 spatio-temporal configurations at level 2, 88 at level 3, 778 at level 4… Note that this representation is compatible with the Allen's time interval relationships. In a more general way, it is also possible to describe life and motion configurations without using the property of continuous spatio-temporal history. In this case, an object can appear and disappear more than one time during the analysis, e.g. workers leave every night their monitored working building and come back every morning. In this case, the number of spatio-temporal configuration blows, e.g. for level 2 there is 100 configurations, 731 for level 3, 4435 for level 4, 24171 for level 5 and 127269 for level 6 … The exhaustive list of the life and motion configurations until level 6 is available at http://www.geo.ulg.ac.be/hallot. The figure 5 presents some extract of the life and motion configurations set. Dashed lines represent non present states.

## 4 Spatio-temporal Generalised Model

Although there are a finite number of life and motion configurations regarding spatio-temporal states, it is still huge and rather difficult to reason with. We propose to generalize life and motion configurations into line-line topological relationships by projecting them into a primitive space. A primitive space is a spatio-temporal space where the spatial and the temporal dimensions are not differentiated [14, 20]. Our aim is to associate spatio-temporal meaning to these generalised spatio-temporal relationships (i.e. the

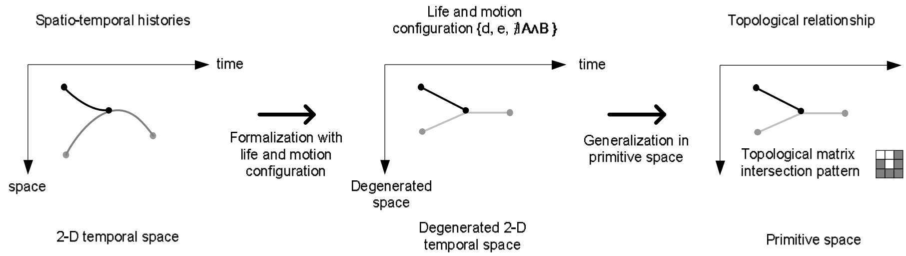



**Fig. 5.** From left to right, formalization of ST histories into a life and motion configuration and generalization of life and motion configuration into topological relationships through a primitive space

topological relationships between spatio-temporal histories in a primitive space) and take advantage of the existing calculus and operators already developed for such topological relationships. When the life and motion configurations are projected from the 2-dimensional degenerated temporal space into a primitive space, spatial and temporal dimensions become undifferentiated. Thus, it is possible to use a topological calculus on the spatio-temporal histories (through their corresponding life and motion configurations) to extract information from the life-lines. We switch from spatio-temporal analysis of a moving point to analysis of lines in a two dimensional space. The figure 5 presents the generalization processes from the formalization of spatio-temporal histories with life and motion configuration to their projection in a primitive space. Topological relationships are represented through their corresponding topological matrix intersection patterns [21].

From [5], we know that there are 33 topological relationships between two lines in a 2-dimensional space. Here, 8 of them are impossible (see fig. 7) and lead us to

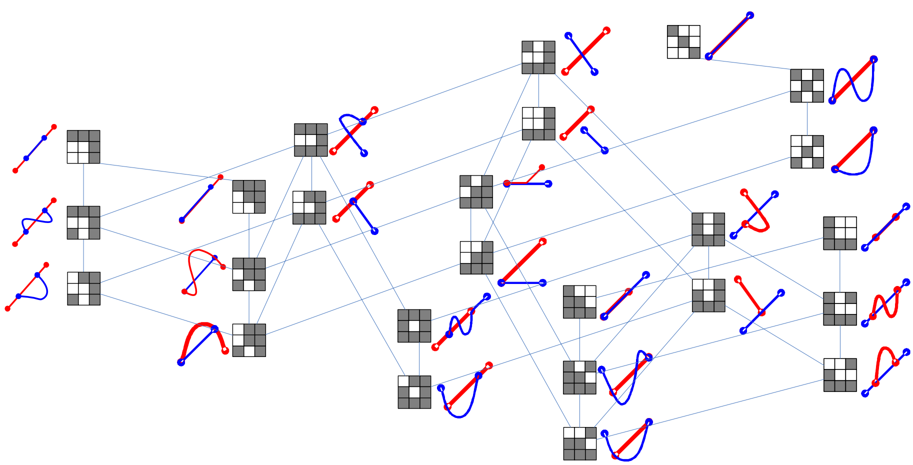


**Fig. 6.** Conceptual neighbourhood diagram of the twenty-five generalised life and motions configurations (inspired from [21])

25 generalised spatio-temporal relationships. The level of generalisation is high. Obviously, part of spatio-temporal information is lost during the generalisation process. We assume that generalized spatio-temporal relationships contain enough meaning to perform efficient spatio-temporal analyses. The full description of the generalization process is out of the scope of this article. The 25 topological intersection matrixes obtained from the generalization are represented into the figure 7. Topological relationships are represented through their topological intersection matrix pattern and with a sample relationship.

Egenhofer and Herring [5] defined 15 conditions reducing the number of possible relationships between to lines to 33. We defined new condition which is necessary for representing generalized spatio-temporal relationships. This condition expresses that points can not move backward in time and that they can not instantaneously move in space. It reduces the number of possible generalized spatio-temporal relationships to 25 (see fig 7).

**Condition:** If A's boundaries not intersect B's exterior, then B's boundaries may not intersect A's interior and vice-versa.

$\mathbf{R}_{\{\emptyset,\neg\emptyset\}}(\mathbf{A},\mathbf{B}) \neq$

| A/E | $A^\circ$ | $\partial A$ | $A^-$ |
|---|---|---|---|
| $E^\circ$ | | | |
| $\partial E$ | $\neg\emptyset$ | | |
| $E^-$ | | $\emptyset$ | |

∨

| A/E | $A^\circ$ | $\partial A$ | $A^-$ |
|---|---|---|---|
| $E^\circ$ | | $\neg\emptyset$ | |
| $\partial E$ | | | $\emptyset$ |
| $E^-$ | | | |

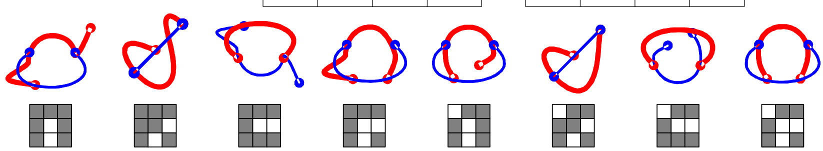


**Fig. 7.** Eight impossible topological relationships between spatio-temporal histories projected in a primitive space

Some recent works including [16, 22] has shown the importance of associating to configurations and relationships common sense interpretation in natural language. It is necessary to improve their understanding and facilitate their integration into intelligent systems. The next figure presents a natural language interpretation of the 25 generalised spatio-temporal relationships (represented by their topological intersection matrix pattern). Theses interpretation may also be retrieve by analysing the topological intersection matrices.

Currently, generalized spatio-temporal relationships model is only available for objects with continuous presence, i.e. continuous spatio-temporal history. Next step of the research will be to propose a generalization including cases of discontinuity presence. Further challenges will be to develop transitions tables and to confront our model with real valid cases studies. We believe that this model could be useful for application such as crime mapping giving shortly information's such as: "Do theses persons meet?", "Did they share time together?", "Has a person ever left another?"…

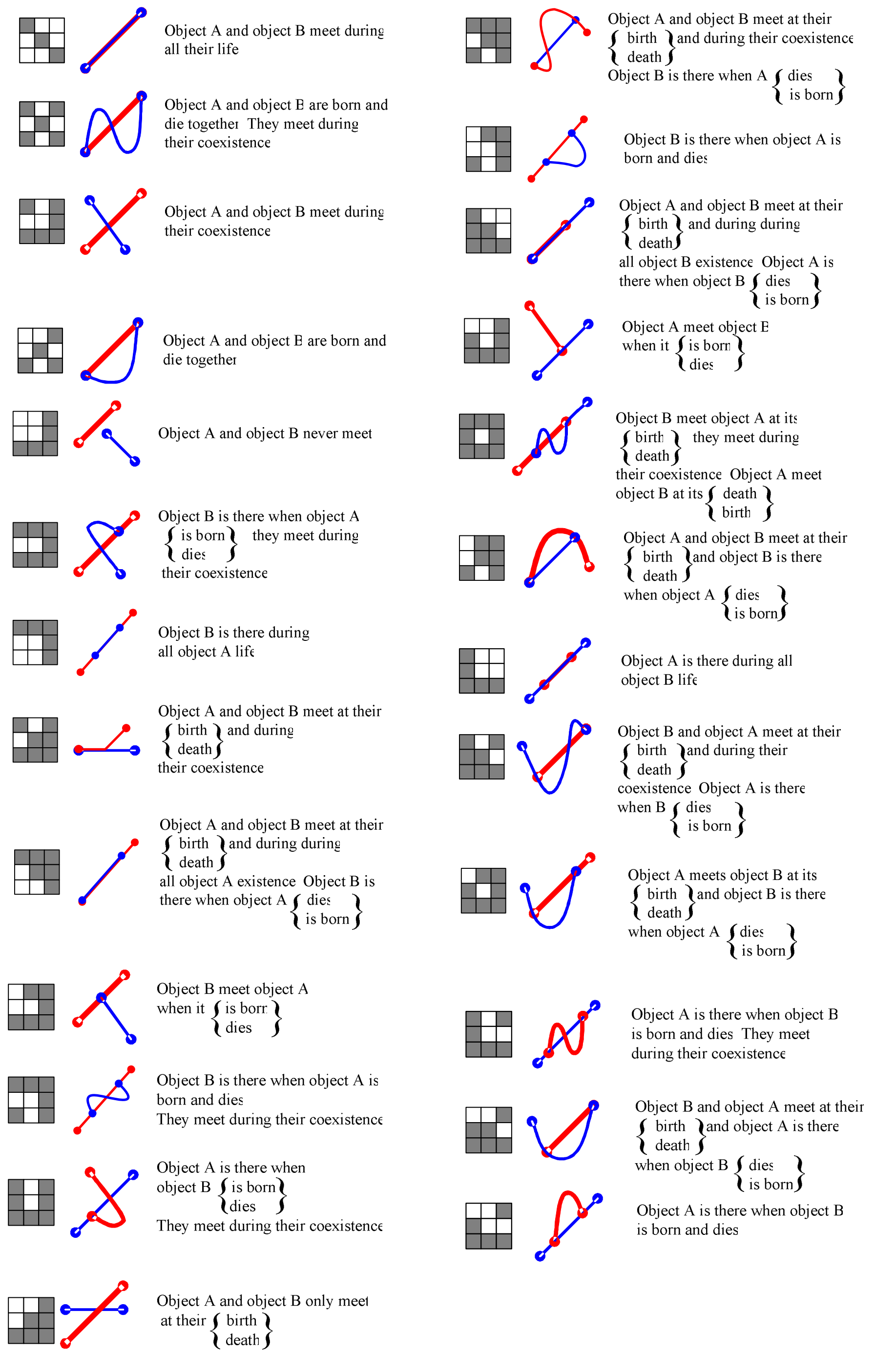



**Fig. 8.** Commonsense interpretation in natural language of the twenty-five generalized of life and motion configurations

## 5 Conclusion

Spatio-temporal reasoning models are needed to fully exploit increasing available spatio-temporal data. These models can be either combinations of spatial and temporal logic into a spatio-temporal logic or based on new mereotopology analysing spatio-temporal shapes. Among others, our model has been developed following the latter way. Some of them, dealing with qualitative trajectory analyses and movement description, are very efficient to describe movement between two disjoint coexisting objects. However, they imply new operators (e.g. in STDBMS or STIS) and cannot deal with not existing and not present objects. Our model aimed to overcome these limitations. First, we have proposed an extended temporal representation of existence and presence of objects. We have used this representation combined with topological concepts between objects to define spatio-temporal states, i.e. *particular spatio-temporal relationships between two objects at a given time*. We ended up with ten possible states which are a JEPD set. Based on this new representation, life and motion configurations, i.e. all the possible interaction between two points in a topological and temporal point of view, have been created to represent formally spatio-temporal histories. This exhaustive set was a strong basis to build our spatio-temporal generalized reasoning model. Indeed, the main idea of our research was to project life and motion configurations into a primitive space, i.e. where spatial and temporal dimensions are not differentiated, and to use topological calculus between lines to extract information. This operation gives us a set of 25 generalized spatio-temporal relationships. We have showed that out of the 33 spatial topological relationships between lines, 8 were impossible for representing spatio-temporal information. Finally, we have proposed a commonsense interpretation in natural language of the generalized spatio-temporal relationships. Future researches will be to check our model‘s validity with real dataset. Note that its implementation should be straightforward as topological operators between lines are implemented in every S(T)DBMS. Among other future developments, integration of other line-line calculi should be envisaged to manage more general cases where life-lines are not continuous.

## References

1. Goodchild, M., Egenhofer, M., Kemp, K.K., Mark, D.M.: Introduction to the Varenius project. International Journal of Geographic Information Systems 13, 731–745 (1999)
2. Hazarika, S., Cohn, A.G.: Qualitative spatio-temporal continuity. In: Montello, D.R. (ed.) Proceedings of Conference On Spatial Information Theory, pp. 92–107 (2001)
3. Freksa, C.: Qualitative Spatial Reasoning. In: Mark, D.M., Frank, A.U. (eds.) Cognitive and Linguistic Aspects of Geographic Space, pp. 361–372. Kluwer Academic Publishers, Dordrecht (1991)
4. Randell, D.A., Cui, Z., Cohn, A.G.: A Spatial Logic Based on Regions and Connection. In: Nebel, B., Rich, C., Swartout, W. (eds.) Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference, pp. 165–176. Morgan Kaufmann, San Mateo (1992)
5. Egenhofer, M., Herring, J.: Categorizing Binary Topological Relations Between Regions, Lines and Points in Geographic Databases. Technical Report. Department of Surveying Engineering, University of Maine, p. 28 (1990)

6. Kontchakov, A., Kurucz, A., Wolter, F., Zakharyaschev, M.: Spatial logic + temporal logic =? In: Aiello, M., Pratt-Hartmann, I., van Benthem, J. (eds.) The Logic of Space, p. 72. Kluwer, Dordrecht (2007)
7. Wolter, F., Zakharyaschev, M.: Spatio-temporal representation and reasoning based on RCC-8. In: Seventh Conference on Principles of Knowledge Representation and Reasoning, KR 2000, pp. 3–14. Morgan Kafmann, Breckenridge (2000)
8. Gerevini, A., Nebel, B.: Qualitative Spatio-Temporal Reasoning with RCC-8 and Allen's Interval Calculus: Computational Complexity. In: ECAI 2002, pp. 312–316. IOS Press, Amsterdam (2002)
9. Cohn, A.G., Bennett, B., Gooday, J.M., Gotts, N.M.: Qualitative Spatial Representation and Reasoning with the Region Connection Calculus. Geoinformatica 1, 275–316 (1997)
10. Allen, J.F.: Towards a general theory of action and time. Artificial Intelligence 23, 123–154 (1984)
11. Mamma, Z., Pnueli, A.: The temporal logic of reactive and concurrent systems. Springer, Berlin (1992)
12. Van de Weghe, N.: Representing and Reasoning about Moving Objects: A Qualitative Approach (Volume I). Department of Geography - Faculty of Sciences. Ghent University, Ghent, p. 168 (2004)
13. Claramunt, C., Jiang, B.: A representation of relationships in temporal spaces. In: Atkinson, P., Martin, D. (eds.) Innovations in GIS VII: GeoComputation, vol. 7, pp. 41–53. Taylor & Francis, London (2000)
14. Muller, P.: Topological Spatio-Temporal Reasoning and Representation. Computational Intelligence 18, 420–450 (2002)
15. Van de Weghe, N., Cohn, A.G., De Tre, G., De Maeyer, P.: A Qualitative Trajectory Calculus as a Basis for Representing Moving Objects in Geographical Information Systems. Control and Cybernetics 35 (2006)
16. Noyon, V., Devogele, T., Claramunt, C.: A formal model for representing point trajectories in two-dimensional spaces. In: Akoka, J., Liddle, S.W., Song, I.-Y., Bertolotto, M., Comyn-Wattiau, I., van den Heuvel, W.-J., Kolp, M., Trujillo, J., Kop, C., Mayr, H.C. (eds.) ER Workshops 2005. LNCS, vol. 3770, pp. 208–217. Springer, Heidelberg (2005)
17. Galton, A.: A qualitative approach to continuity. In: Amsili, P., Borillo, M., Vieu, L. (eds.) Proceedings of the 5th International Workshop on Time, Space and Movement: Meaning and Knowledge in the Sensible World (TSM 1995), Toulouse, France, pp. 17–30 (1995)
18. Hayes, P.: Naive physics I: ontology for liquids. Readings in qualitative reasoning about physical systems, pp. 484–502. Morgan Kaufmann Publishers Inc., San Francisco (1990)
19. Hallot, P., Billen, R.: Spatio-Temporal Configurations of Dynamics Points in a 1D Space. In: Gottfried, B. (ed.) Behaviour Monitoring and Interpretation BMI 2007, Centre for Computing Technologies (TZI), University of Bremen, Germany, Osnabrück, Germany, pp. 77–90 (2007)
20. Muller, P.: Éléments d'une théorie du mouvement pour la formalisation du raisonnement spatio-temporel de sens commun. Institut de recherche en informatique de Toulouse, p. 219. Université Paul Sabatier, Toulouse (1998)
21. Kurata, Y., Egenhofer, M.: The 9+-intersection for Topological Relations between a Directed Line Segment and a Region. In: Björn, G. (ed.) Workshop on Behaviour Monitoring and Interpretation (BMI 2007), University of Brement, TZI Technical Report, Osnabrück, Germany, vol. 42, pp. 62–76 (2007)
22. Bandini, S., Mosca, A., Palmonari, M.: Commonsense Spatial Reasoning for Context–Aware Pervasive Systems. Location- and Context-Awareness, 180–188 (2005)

# A Semantic and Language-Based Model of Landscape Scenes

Jean-Marie Le Yaouanc, Éric Saux, and Christophe Claramunt

Naval Academy Research Institute,
BP 600, 29240, Brest Naval, France
{leyaouanc,saux,claramunt}@ecole-navale.fr

**Abstract.** The modelling of landscape environment is a cognitive activity, that still requires novel kinds of spatial representations. This paper introduces a structural categorisation of a landscape view based on panoramic photographs that act a substitute of a given natural environment. Verbal descriptions of a landscape scene provide the modelling input of our approach. The structural-based representation identifies the spatial, relational and semantic constructs that emerge from these descriptions. A landscape view is modelled according to the structure of its language description. Concepts in the environment are qualified according to an ontological classification, their proximity to the observer, and the relations that qualify them. The resulting model is schematised by a music score-based representation that constitutes a modelling support for the study of environmental descriptions.

**Keywords:** Landscape perception, spatial cognition, verbal descriptions.

## 1 Introduction

Early developments of Geographical Information Systems have been widely influenced by quantitative representations of space. However, it has been recognised that quantitative models do not reflect the way humans perceive, conceptualise and describe their environment [1]. Over the past few years, novel approaches based on spatial cognition and qualitative perceptions of space have emerged as alternative solutions. In particular, the development of wayfinding and landscape representations have emphasised the potential and role of verbal descriptions in the modelling of large-scale spaces, including the way landmarks and spatial relations are formalised [2]. It has also been long recognised that natural language favours the understanding on how people conceptualise space [3].

The research presented in this paper addresses the conceptualisation and the formalisation of a panoramic natural landscape perceived by an observer, and its materialisation by a verbal description. Our study considers a verbal description of an environmental scene as the initial context of the modelling approach whose objective is to categorise the constructs and semantics exhibited. These environmental scenes consist of 360$^\circ$ panoramas materialised by photograph montages presented to the observer with a dynamic interface. These photographs acts as

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 334–343, 2008.

a substitute of a given natural environment. The verbal descriptions considered are extracted from a panel of observers who are given the task of depicting these views in a way that allows recognition of the scene by an external addressee. The study focuses on the identification and categorisation of the spatial and structural constructs that emerge from these scenic perceptions. The modelling purpose of the research is to formalise the way a panoramic scenic environment is qualified by a verbal description. The forms that emerge from our modelling approach should help identify the structure, rhythm, patterns and properties of a scenic description. The schematisation results in a two-dimensional structural representation, based on the notions of proximity spaces, and a linear ordering of the verbal descriptions.

The remainder of the paper is organised as follows. Section 2 briefly introduces related work on the cognitive representations of natural landscapes. Section 3 presents our experimental study, where a panel of observers are given the task of producing verbal descriptions. Sections 4 and 5 provide a conceptualisation of these verbal descriptions. Finally, section 6 draws the conclusions and outlines further work.

## 2 Landscape Perception

It has been shown that many important features in landscape descriptions cannot be observed by conventional cartographical representations. Photographs have been recognised as substitutes for *in-situ* landscape observations as they facilitate the acquisition of verbal descriptions [4]. Although photographs have limited capabilities for the identification of small or detailed elements due to the absence of depth, they provide a valuable support for a semantic interpretation of descriptive observations.

Cognitive studies usually distinguish two levels in the mechanisms and abstractions associated to mental interpretation. On the one hand, the internal level, focussing on the neuropsychological process, which generates a mental representation. On the other hand, the external level which reproduces the structural organisation of the observed phenomenon, for instance at the language level. Our research concerns the latter. As it has been shown in early studies, an observer acting in a given environment, organises space according to proximities and senses [5,6]. The “proximate environment” is defined as the area perceived by the senses, while the “landscape area” is perceived by sight alone. The landscape area extends to the horizon separating the earth from the sky. The notion of mental spaces has been introduced and “schematised by eliminating detail and simplifying features around a framework consisting of elements and the relations among them” [7]. Tversky makes the difference between “the space of the body”, “the space around the body”, “the navigational space” and “the space of graphics” [8]. Similarly, Montello categorises the “figurative”, “vista”, “environmental” and “geographical” space [9]. These cognitive studies clearly show that the perception of environment is organised at different levels, according to the perception and the behaviour of the observer.

Studies on the perception of distances show evidence that proximity to concepts composing an environment is particularly influenced by salient features. Concepts act as references when they are salient and permanent in the environment [6]. The term *concept* includes regions, parcels of land and water bodies, structural features such as mountains, hills or valleys, and human-made features such as roads or buildings. Although they are not always precisely located in the environment, concepts often serve as frames of reference. This is largely due to the nature of concept boundaries and the distinction made between "fiat boundaries", *i.e.*, the one defined by human demarcation, and "bona fide boundaries", *i.e.* the ones that refer to physical discontinuities, such as a mountain, valley or meadow [10].

## 3 Experimentation

This study has been carried out in the context of a semi-natural landscape located in the Alpine region of France. The experimentation was conducted using a panel of 22 participants, 17 males and 5 females, mainly non-experts in spatial cognition studies. After a quick overview of the panoramic photograph using a dynamic interface[1], they were asked to spontaneously describe the photograph for an external addressee whose goal was to recognise the scene. Verbal descriptions were recorded using a data storage device and were not to exceed 5 minutes.

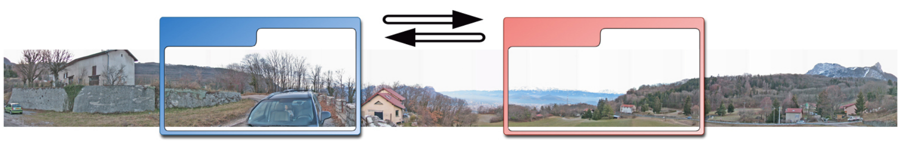


**Fig. 1.** Experimental panorama of a semi-natural environment

The 360° panoramic images were displayed on a computer screen. The photographs were presented interactively and non interactively. When the user chose a non-interactive method, the panorama was displayed with a constant rotational speed. The panoramic view was also presented in an interactive manner: the observer was able to explore the scene by rotating the view, just as he would have done it by rotating his body in a natural environment. Photographs were displayed in a large size in order to limit the restriction due to the difference of size between an environment perceived *in situ*, and the one perceived through a photograph.

An example of description that emerges from the landscape presented in figure 1 is as follows: *"I'm in a mountainous region. In front of me, there is a house with a little garden. 500 meters behind the house, a footpath crosses a large meadow. Far away, behind the meadow, I can see a huge town in a valley.*

[1] The experimental setup is available at `http://experimentation.yaou.org/`

*On the horizon, there is a mountain range. Behind me, there is a large road that runs around a hill and a footpath winding around it. There is a little cottage along the road, and crossroads on the right of the cottage."*

The verbal descriptions generated by the panellists have led the following results. Most of the participants describe the panoramic photograph from left to right, taking a reference point to begin their description. Most verbal descriptions are implicitly organised with a hierarchy. 60 % of the participants first describe the scene as a whole with sentences like "I am in a mountainous region". The salient concepts structuring the scene are first mentioned but not localised. Next, participants describe the content of the landscape, and the way these concepts are related to each others. This shows evidence of a hierarchical perception of space where the landscape is first perceived and described as a whole, before being detailed. This confirms previous studies and evidence of hierarchies in the perception of spatial information [11]. When observing the environment, humans perceive distinct concepts that are part of the landscape. Most descriptions attempt to identify this variety of concepts which are directly related to the cultural and ontological background, and common sense of the observer. The scene description is largely based on salient concepts, as their role is prominent in the structure and organisation of the environment [12]. Concepts are qualitatively associated to others using spatial relations, preferably with landmarks as frames of reference. With respect to the considered environment, participants used man-made (50% of the identified entities), relief (30%) and vegetation (15%) concepts, where landmarks are assimilated to punctual entities, and forms to extended entities. The salience and legibility of these concepts largely influence their role in the perception process.

The observers interpret photographs of an environmental scene by using spatial relations to qualify the position of the concepts such as "behind the mountain", "in front of the house", "in the background", "in the foreground", "in the long distance", proximity adjectives such as "near", "close to", "far from", "further", directional relations such as "to the right of" and a few tridimensional constructs such as "above" or "below". These concepts refer to "image schematas" used in the perception of large-scale spaces [13]. These terms are either egocentric constructs and vary depending on the observer's location, or relative constructs associated to the location of another concept identified in the landscape. This shows the complementary roles played by egocentric and relative frames of reference in scene perception. It also appears that the roles played by the frames of reference selected by the observer, are determined by their contribution to the hierarchical organisation of the environment. This confirms Tversky's intuition on the organisation of environment that impacts the structure of a verbal description and precedes its linear structure [14]. The fact that different frames of reference are used within a description may also reflect the fact that speakers tend to use a greater variety of terms and rhythms which, in turn, generate the richness of the description. The location of these concepts depends on their proximity to the observer. Proximate spaces which emerge from these descriptions, vary according to the terms used and the spatial extent of the concepts.

The experiment has also highlighted the prominent role played by direction relations (50%) which are twice as often used as proximity (30%) and topological constructs (20%). This stresses the fact that direction relations are among the most appropriate in structuring a panoramic view. It is also worth noting the low role played by quantitative measures in the descriptions although two participants used metric data in their description to specify a proximity. Moreover, these quantitative measures are associated to a fuzzy linguistic term such as “about 500 meters from my position” or “around 400 meters from the house”.

## 4 Modelling Approach

We define an *environmental scene* as the 360$^\circ$ environment perceived and described by an observer from a static point of view. An environmental scene is composed of a combination of concepts associated by using spatial relations. These concepts are perceived as forms in the landscape *i.e.*, mountains, hills, vegetal and cultivated areas, or as landmarks, *i.e.* buildings and all other man-made features, salient in the environment. A landscape scene is qualitatively divided into proximity spaces which are defined according to their distance to the observer. These proximity spaces correspond to the ones visually accessible to the observer. Our classification also considers the space on the horizon in order to mark the boundary between the environment accessible to the observer, and the geographical space beyond the line of sight. These proximity spaces are defined as follows:

- The *space around the body* as introduced by Tversky [8] that is described by terms such as “near me”.
- The *experienced space* that can be easily experienced through locomotion. It is described by terms such as “not so far”.
- The *distant space* is the environment, between the experienced space and the space on the horizon. It is is described by terms such as “far away”.
- The *space on the horizon* is the scene in the far distance, and made of silhouettes that constitute landform boundaries. It is described by terms such as “in the background” or “on the horizon”.

A verbal description is made of a corpus of sentences. A sentence contains several concepts associated using relations that create the structure of the scene. We characterise a sentence by the concepts, *i.e.*, forms and landmarks, modelled as entities, and the relations that relate them. More formally, a verbal description D is formalised as an ordered set of sentences $s_i \in \mathbb{S}$, the set of sentences, *i.e.*, $D = [s_1, s_2, \ldots, s_n]$. A sentence is an ordered set of entities and relations where every relation has an entity as a predecessor, and an entity as a successor. Let $\mathbb{E}$ be the set of entities and $\mathbb{R}$ the set of relations, a sentence $s_i$ is defined as follows:

– $s_i = [t_1, t_2, \ldots, t_n]$ with $t_1, t_2, \ldots, t_n \in \mathbb{E} \cup \mathbb{R}$ such as
– $\forall t_j \in s_i / t_j \in \mathbb{R}$ then $\exists t_{j-1}, t_{j+1} \in s_i / t_{j-1}, t_{j+1} \in \mathbb{E}$

Let us consider the sentence “500 meters behind the house, a footpath crosses a large meadow” taken from the example introduced in section 3. This sentence is formalised as $s_i$ = [$t_1$ =“500 meters”, $t_2$ =“behind”, $t_3$ =“house”, $t_4$ =“footpath”, $t_5$ =“crosses”, $t_6$ =“large meadow”]. Entities and relations are instanced, labelled by attributes and associated to a semantics (figure 2).

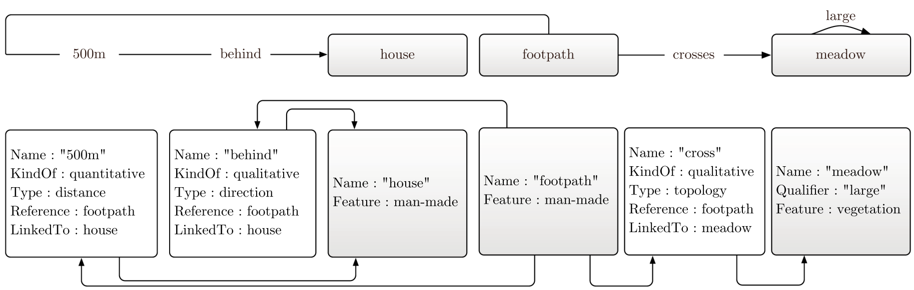



**Fig. 2.** Sentence description principle

## 5 Schematisation of a Verbal Description

This modelling approach constitutes a preliminary logical and structural representation of the verbal description of a landscape scene. The next objective is its schematisation in order to facilitate the understanding of the concepts and structures that appear in the description. The main notions that play an important role are the:

– hierarchical and linguistic properties that emerge from the sentences,
– proximity spaces,
– concepts, *i.e.*, forms and landmarks,
– relations, *i.e.*, direction, proximity and three-dimensional constructs,
– ontology references that classify the identified forms and landmarks.

We consider that a verbal description is ordered by two orthogonal dimensions: the boundaries and rhythm given by the sentences, and the proximity spaces. This leads us to retain a graphical representation designed by analogy to scores used in music, and where notes rhythm the time line, and where the pitch and duration characterise every note. Within our representation, the rhythm is given by the sentences, forms, landmarks and relations. Proximity spaces model the pitch of every concept in the landscape, its extent and its duration on the music score-based notation (figure 3). The following subsections illustrate the instantiation of the ontological and music score-based schematisation.

### 5.1 Ontological View

The ontological view is a semantic schematisation linked to the properties of the forms and landmarks that appear in the verbal description. Concepts and relations are represented by their linguistic terms associated to a semantic derived from the ontology of foot-orienteering. This ontological view combines a conceptual diagram with a proximity-based structural representation, and a phrasing structure that reproduces the rhythm of the verbal description (figure 3). This example shows that different frames of reference, located in different proximity spaces, are used in the scenic description (e.g, the observer, the house, the meadow, the valley and the road).

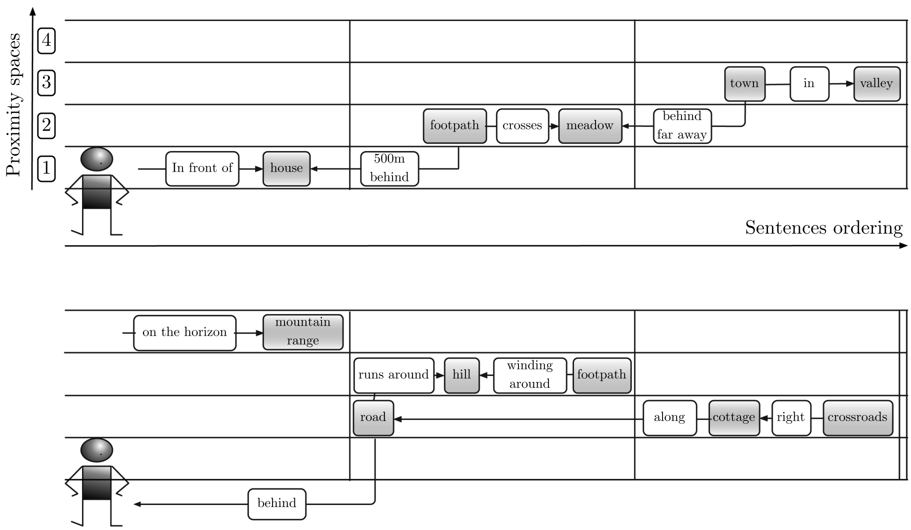



**Fig. 3.** Ontological view of a verbal description with the proximity spaces. 1. is the space around the body, 2. is the experienced space, 3. is the distant space and 4. is the space on the horizon.

### 5.2 Music Score-Based View

The music score-based view is an interpretation of the ontological view where concepts are preferably represented by their extent in space, by similarity to the way notes are time-stamped on a music score (figures 5 and 6). With respect to the different levels of proximity spaces, concept symbols are appropriately located and indicate their distance to the observer. Note values show the spatial extent of the concepts rather than the relative duration as it is the case with a musical notation. For example, a one dimensional concept is represented by a sixteenth note as it is punctual in space. Similarly, the half note refers to a valley and the whole note to a chain of mountains. As the boundaries of proximity spaces are not always well-defined, the concepts identified are likely to extend over several proximity spaces.

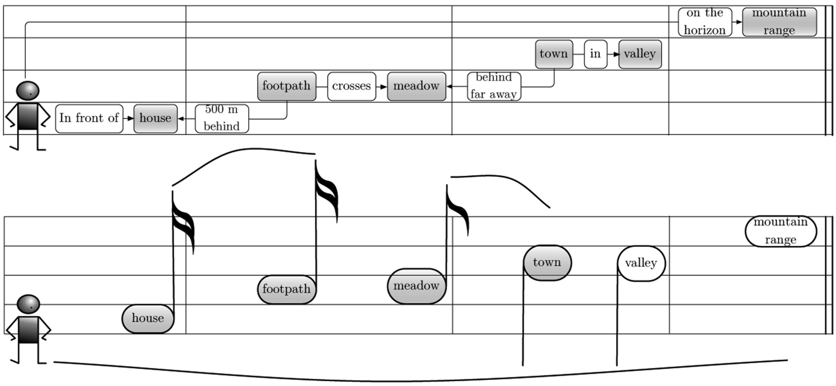
Not surprisingly, the analogy with a music score is not complete. With a musical notation, a bar is a segment of time defined as a given number of beats of a given duration. This constraint is not respected by the music score-based view as sentences do not contain a regular quantity of information. This is due to the fact that the number of landmarks and forms contained in a two-dimensional space is practically not constraint, as it is in a one-dimensional space. The rhythm that emerges from this representation reflects the richness of the sentences, and overall, of the verbal description. Figures 4, 5, and 6 provide two scenes, with examples of ontological and music score-based views derived from a linguistic description produced by an observer. It appears that this observer first describes nearby, and then distant features. We also observe that the rhythm diversity, and the number of sentences exhibited by scene 1 is higher than the one of scene 2. This reflects the fact that the observer has perceived scene 1 as semantically richer than scene 2.

(a) Scene 1 (b) Scene 2

**Fig. 4.** Studied environmental scenes

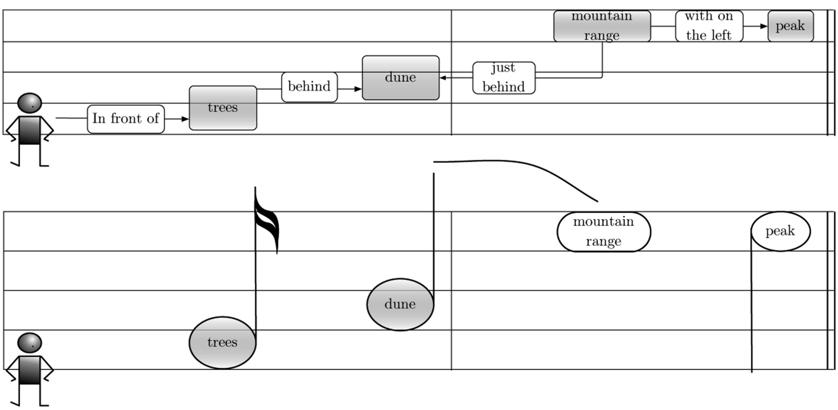
The ontological and music score-based representations can be combined in order to provide a global schematisation of an environmental scene. The approach provides a mean for the analysis of differences and similarities in the linguistic descriptions of landscape scenes.

The formalism is adapted to the identification of the rhythm given by the structure of the sentences, and the depth of the landscape as reflected by the properties of identified concepts. The formalism and its graphical expression can be used for analysing similarities and differences across either a same or different landscape scenes. This gives a methodological support for ethnophysiography studies where the objective is to characterise linguistic and cultural patterns used in landscape perception [15]. It also provides a preliminary resource for the development of a mapping between a verbal description and a computational representation. This formalism constitutes a complementary representation to conceptual and spatial models oriented to the semantics of geographical data.




*In front of me, there is an house. 500 meters behind the house, a footpath crosses a large meadow. Far away, behind the meadow, I can see a town in a valley. On the horizon, there is a mountain range.*

**Fig. 5.** Ontological and music score-based views - scene 1




*In front of me, I can see a few trees and an huge dune behind them. Just behind the dune, there is a mountain range with a peak on the left.*

**Fig. 6.** Ontological and music score-based views - scene 2

## 6 Conclusion

This paper introduces a language-based and cognitive approach that models a verbal description of a landscape scene. A verbal description is modelled by a structural and proximity-based representation that reflects its semantics. The structural properties of a verbal description are linked to a music score-based schematisation, where sentences are characterised by staves, and proximity spaces by their distance from the observer. Ontological-based and music score-based shematisations provide a symbolic view of a landscape scene.

This approach relies on a semantic-based analysis of verbal descriptions as used in landscape perception. It potentially allows cross-comparison of different types of descriptions and environments, comparing their structures, the patterns that emerge, and the different terms and relations used. Further work concerns an extension of the ontological background of the model, and additional experimentations in different natural environments and user contexts.

## References


1. Egenhofer, M., Mark, D.: Naive Geography. In: Frankand, A.U., Kuhnand, W. (eds.) COSIT 1995. LNCS, vol. 988, pp. 1–15. Springer, Heidelberg (1995)
2. Brosset, D., Claramunt, C., Saux, E.: A location and action-based model for route descriptions. In: Fonseca, F., Rodrígues, M.A., Levashkin, S. (eds.) GeoS 2007. LNCS, vol. 4853, pp. 146–159. Springer, Heidelberg (2007)
3. Mark, D., Franck, A.U.: Cognitive and Linguistic Aspects of Geographical Space. Behavioural and Social Sciences, vol. 63. Springer, Heidelberg (November 1991)
4. Shuttleworth, S.: The use of photographs as an environment presentation medium in landscape studies. Journal of Environmental Management 11(1), 61–76 (1980)
5. Granö, J.: Reine Geographie. Acta Geographica, vol. 2(202) (1929); Tr. as Pure Geography, reprinted in English from Johns Hopkins Press (1997)
6. Tversky, B., Lee, P.U.: How space structure language. In: Freksa, C., Habel, C., Wender, K.F. (eds.) Spatial Cognition 1998. LNCS (LNAI), vol. 1404, pp. 157–175. Springer, Heidelberg (1998)
7. Mainwaring, S., Tversky, B., Ohgishi, M., Schiano, D.: Descriptions of simple spatial scenes in English and Japanese. Spatial Cognition and Computation 3(1), 3–42 (2003)
8. Tversky, B.: Cognitive maps, cognitive collages and spatial mental models. In: Frank, A.U., Campari, I. (eds.) COSIT 1993. LNCS, vol. 716, pp. 14–24. Springer, Heidelberg (1993)
9. Montello, D.: Scale and multiple psychologies of space. In: Frank, A.U., Campari, I. (eds.) COSIT 1993. LNCS, vol. 716, pp. 312–321. Springer, Heidelberg (1993)
10. Smith, B., Varzi, A.: Fiat and bona fide boundaries: towards an ontology of spatially extended objects. In: Hirtle, S.C., Frank, A.U. (eds.) COSIT 1997. LNCS, vol. 1329, pp. 103–119. Springer, Heidelberg (1997)
11. Hirtle, S., Jonides, J.: Evidence of hierarchies in cognitive maps. Memory & Cognition 13(3), 208–217 (1985)
12. Couclelis, H.: Worlds of information: the geographic metaphor in the visualization of complex information. Cartography and Geographic Information Systems 25(4), 209–220 (1998)
13. Franck, A.U., Raubal, M.: Formal specification of image schemata – A step towards interoperability in geographic information systems. Spatial Cognition and Computation 1(1), 67–101 (1998)
14. Tversky, B.: Narratives of space, time, and life. Mind & Language 19(4), 380–392 (2004)
15. Mark, D.M., Turk, A.G.: Landscape categories in Yindjibarndi: ontology, environment, and language. In: Heidelberg, S.B. (ed.) COSIT 2003. LNCS, vol. 2825, pp. 28–45. Springer, Heidelberg (2003)

# An Ontology-Based Approach for the Semantic Modelling and Reasoning on Trajectories

Miriam Baglioni[1], José Macedo[2], Chiara Renso[3], and Monica Wachowicz[4]

[1] Department of Computer Science, Univ. of Pisa, Italy
baglioni@di.unipi.it
[2] EPFL, Lausanne, Switzerland
jose.macedo@epf.ch
[3] KDDLAB, ISTI-CNR, Pisa, Italy
chiara.renso@isti.cnr.it
[4] Polytechnic University, Madrid, Spain
m.wachowicz@topografia.upm.es

**Abstract.** In this paper we present a methodology for the semantic enrichment of trajectories. The objective of this process is to provide a semantic interpretation of a trajectory in term of behaviour. This has been achieved by enhancing raw trajectories with semantic information about moves and stops and by exploiting some domain knowledge encoded in an ontology. Furthermore, the reasoning mechanisms provided by the OWL ontology formalism have been exploited to accomplish a further semantic enrichment step that puts together the different levels of knowledge of the domain. A final example application shows the added power of the enrichment process in characterizing people behaviour.

## 1 Introduction

A flood of data pertinent to moving objects is available today, and will be more in the near future, particularly due to the automated collection of telecommunication data from mobile phones and other location-aware devices. These floods of data enable a novel class of applications, where the discovery of consumable, concise, and applicable knowledge is the key step. The presence of a large number of location-aware wirelessly connected mobile devices presents a growing possibility to access *space-time trajectories* of these personal devices and their human companions: trajectories are indeed the traces of moving objects and individuals. These mobile trajectories offer interesting practical opportunities to find *behavioural patterns*, to be used in various application settings, ranging from recreational activities monitoring, to traffic and sustainable mobility management.

However, raw space-time trajectories, encoding position coordinates equipped with a timestamp, do not provide any information about the travelling object, geographical space and background knowledge. To be really useful to the end-user, raw trajectory data must undergo to a semantic enrichment process that permits the integration of different kinds of knowledge (spatio-temporal, trajectory, geographic, background, domain, etc) in a seamless and progressive manner.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 344–353, 2008.

Besides, trajectory semantics must be manipulated or extended directly by the user using some expressive language that can be human consumable. With this requirement, we avoid hiding trajectory semantics inside some artefact (e.g. like conceptual data model does using graphical notation). The last but not least requirement is that trajectory semantics must provide support to inference tasks that help user to validate and gain more knowledge about trajectory data. Thus, a good candidate for accomplishing all these requirements is a formal ontology.

Ontologies [10] have certainly become a research topic in several disciplines, ranging from philosophy, geography, geomatics up to machine learning and artificial intelligence. Several approaches have been proposed for the development of formal ontologies. They may differ from each other in terms of methods for defining concepts, for identifying relations between these concepts and for building reasoning tasks [21]. In particular, [4] suggests that semantic models are unique sources for deriving ontologies because they already include appropriate definitions of concepts. However, most of the data models developed so far have considered moving objects with an absolute representation of space, and have proposed query languages and data mining functions that support their manipulation in space and time [11]. Within these semantic models, a moving object is usually represented using its location as a function of time. A different formal model has been proposed by [13] where the basic primitives are the relative positions and relative velocities of the neighbouring objects and regions, as perceived by an observer acting in the environment.

In this paper, an enrichment process is proposed to derive trajectory ontology from the abstraction of a semantic model of trajectories. The objective of this process is to provide a conceptual semantic model centred to a trajectory rather than a moving object of interest. This has been achieved by enhancing raw trajectories with semantic information about moves and stops. Furthermore, the reasoning mechanisms provided by the ontology formalism have been exploited to accomplish a further semantic enrichment step that puts together the different levels of knowledge of the domain.

Structure of the paper follows. Section 2 describes the process of defining semantic enrichment of trajectories . Section 3 defines semantic trajectories as stop and moves and gives an ontological representation. Section 4 presents the application domain that exemplifies the approach and ontology reasoning. Conclusions and open issues are reported in Section 5.

## 2 The Semantic Enrichment Process

Trajectories captured by positioning systems (e.g. GPS and GSM equipments) suffer from a lack of semantics. Data acquired by those mechanisms are usually represented in a form of a sequence of *(sample point, time)* pairs, which rigorously reveals the geometric facet of a trajectory (called raw trajectory). Actually, a geometric facet is a limited concept to describe the semantics of trajectories because it does not provide any information about the travelling object, geographical space and background knowledge. Therefore, there is a need for enhancing raw trajectories with semantic information.

The semantic enrichment process we propose starting from the geometric representation of the trajectory of an individual, enriches it with contextual information in

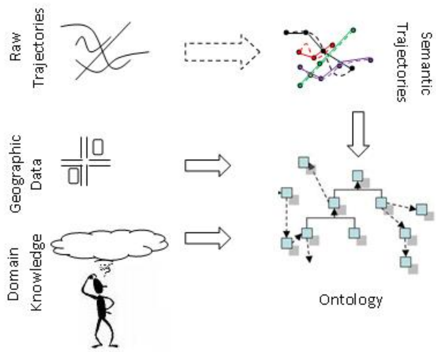



**Fig. 1.** The Semantic Enrichment Process

terms of geography and domain knowledge. The obtained semantic trajectories can be easily represented and integrated in a formalism that allows for a further reasoning step capable of characterizing movements in terms of people behaviour. This process is illustrated in Fig.1.

The first step of the semantic enrichment process takes a raw trajectory and produces as output a new trajectory representation where semantic context information is taken into account. Typically, a semantic trajectory will be richer than the original raw trajectory in terms of geographical and application semantics. On the other hand, it has less informative content in terms of geometric details. It is worth noticing that the semantic enrichment step strongly depends on the application and reasoning requirements. Moreover, semantic representation of a trajectory typically is simpler than the geometric trajectory, thus allowing an easier manipulation in terms of knowledge representation language.

In the following sections, we introduce a definition of semantic trajectory that has been proposed by Spaccapietra et al in [20] as *Stop* and *Moves*. However, other definitions could be proposed for semantic trajectories. For example, in an ethological application it would be more interesting to explicitly represent animals trajectory by means of the geographical areas where animals move thorough, than the stops themselves.

The second step of the enrichment process integrates the semantic trajectory representation into a more general ontology where both geographical and application domain knowledge are encoded. As a result, the last step exploits the reasoning capabilities of the chosen representation language to perform automatic inferences that characterize semantic trajectories in terms of *behaviour*, as illustrated in Section 4.

## 3 From Raw Trajectories to Semantic Trajectories

From the user's perspective, trajectory is a semantic spatio-temporal concept because it is viewed as the evolving position of some object travelling in some space during a given time interval, having a specific meaning or goal. Raw and user view of trajectories are not necessarily the same because the segmentation of a raw trajectory path into semantic trajectories depends upon the user's mental model. Thus, a precise definition of semantic trajectory is required in order to allow users to choose their conceptualisation of the raw trajectory data. In [21], a spatio-temporal conceptual model is proposed to provide constructs and rules that enable the designer of a database that uses movement data to think about these data as sets of identifiable trajectories travelled by application objects. Apart from the work cited above, the state of the art of conceptual models for spatio-temporal databases does include the moving point construct e.g. [12, 15]. Another example is [3] where the authors aim at enriching the semantics of a moving object model. However, the trajectory is the polyline connecting the sample points that define the discrete representation of movement, i.e. a sequence of spatio-temporal segments, rather than a sequence of semantic steps.

### 3.1 The Semantic Trajectory: Definition

Here we use the following concepts that settle trajectories as a semantic object as one of the following:

***Definition 1 (Trajectory).*** A trajectory is the user defined record of the evolution of the position of an object that is moving in space during a given time interval in order to achieve a given goal.

***trajectory: [time interval] → space***.

Travelling objects do not necessarily continuously move during a trajectory. Consequently, trajectories may themselves be semantically segmented by defining a temporal sequence of time sub-intervals where alternatively the object position changes and stays fixed. We call the former the moves and the latter the stops. We can then see a trajectory as a sequence of moves going from one stop to the next one (or as a sequence of stops separating the moves). Yet, identifying stops (and moves) within a trajectory is the responsibility of the application.

***Definition 2 (Stop).*** A stop is a part of a trajectory, such that the user has explicitly defined this part of the trajectory to represent a stop. More formally, a stop has a non-empty time interval and the spatial extent is a single point.

***Definition 3 (Move).*** A move is a part of a trajectory delimited by two extremities that represent either two consecutive stops, or trajectory beginning or end. A move is represented by spatio-temporal line (not a point) defined by the trajectory function.

Different levels of semantic specification are necessary in order to provide different levels of interpretation. For example, a very basic semantic concern is supporting the definition of the stops, if any, which decompose the trajectory into a sequence of moves. Another standard semantic concern is giving a meaning to trajectory components, i.e. defining their semantic interpretation in terms of the application objects they represent. Actually, this process of semantic enrichment shall not be limited or rigid, which means that a user must be able to describe its interpretation of the facts by using an expressive language that permits the reuse of predefined semantics.

### 3.2 The Semantic Trajectory: Ontology Representation

A formal representation of concepts requires a "specification of a conceptualisation", according to Gruber's definition of ontology [8]. A formal ontology is based on "a conceptualisation of objects, concepts and other entities that exist in a given domain and their relationships" [9]. They rely upon a logical theory that defines the basic concepts along with their relations, and formal axioms to constrain the possible interpretation of terms. Concepts are defined as taxonomic hierarchies of classes, thus the most important type of relation is the *subsumption*, commonly known as “is a” relationship. Ontology may also include individuals, the ground facts or concrete objects that are instances of a class in a database. The set of individuals represents the knowledge base of a formal ontology.

The semantic conceptualisation of a trajectory is represented as a sequence of Stops connected by Moves. Notice that we have four explicit different relations connecting Stops to Moves, namely, *fromStop*, *toStop*, *fromMove*, *toMove*.

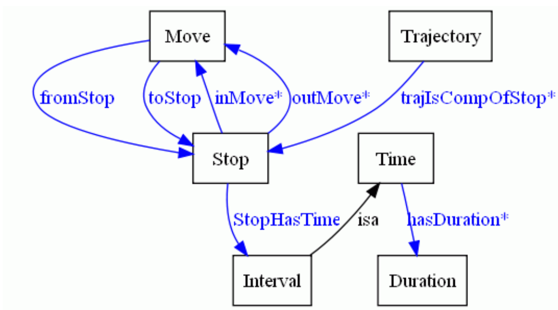



**Fig. 2.** The semantic trajectory representation in the ontology

More precisely, almost every Stop is connected to other two Stops by two Moves. This is graphically shown in Figure 2. Furthermore, every Stop is connected to an interval that represents the duration of the stop.

Actually, a specification of trajectory semantics is application dependent, which means that a generic trajectory semantic characterization is not suitable for all kinds of application. Besides, semantics of trajectories must be contextualised according to users’ mental model. For example, the same child trajectory can be described differently according to his parents or a recreational planner. The parents knowing their child habits may classify their trajectory having dangerous and suspicious behaviour, whereas a recreational planner may interpret this trajectory as a common behaviour.

Therefore, in the following section we introduce an example of application domain on recreational planning, and we show the added value given by the final reasoning step.

## 4 A Recreational Planning Application and Its Ontological Representation

Mobility is an integral part of the process of recreation planning. The latest achievements in the field of mobile networks and ubiquitous computing enable the integration and combination of technologies like Internet, Java, and multimedia in a new kind of applications for outdoor recreation that employs distribution, wireless networking, localisation, and movement – mobile movement-aware multimedia games. The term mobile game typically stands for an isolated interactive game on a mobile device which does not utilise location and movement of the player. The availability of positioning technology such as GPS, together with more powerful mobile devices, and mobile networking infrastructures can allow us to develop mobile location-based multimedia games. Existing mobile games that introduce location in their application context often go into the direction of augmented reality, using extensive hardware and software.

*Paper chase* (also known as *scavenger hunt*) is an old children's game in which an area is explored by means of a set of questions and hints on a sheet of paper. Each team tries to find the locations that are indicated on the questionnaire. Once a place is found, they try to best and quickly answer the questions on the sheet, note down the answer, and proceed to the next point of interest (POI). Based on this game idea, we have used the mobile location-aware game *Frequency 1550* developed by Waag Society, in the Netherlands for students in the age of 12-14. It is a research pilot examining whether it's possible to provide a technology supported educational location-based experience. Students are transported to the medieval Amsterdam of 1550 via a medium that is familiar to this age group: the mobile phone.

The game mainly consists of a set of geo-referenced checkpoints associated with multimedia riddles. With the mobile game client a player logs in to the game server, receives a map and the player has to find checkpoints during the game. Each of checkpoints is transformed into a screen coordinate and drawn on the map. The player's device includes a GPS receiver which continuously tracks the current position of the device. The player tries to solve the riddle not only correctly but also as quick as possible, because the time needed to solve all the riddles is accumulated and added to the overall score. The answer to the riddle is communicated to the game server. It is possible for the player to interact with the system but also with other people, e. g., by asking them for help with the riddles.

One main application requirement has been selected based on the general lack of knowledge of recreation planners about the actual movement of players within a recreation site. The assumption is that game activities are not always performed in planned recreational zones. Thus an analysis of movements would allow a better understanding of how people behave during the game.

### 4.1 Reasoning on Recreational Activities

Ontology languages are formal languages used to construct ontologies. They allow the encoding of knowledge about specific domains and often include reasoning facilities that support the processing of that knowledge. Among all the ontology languages, we considered Web Ontology Language (OWL), that is a well known standard arisen from the Semantic Web and it is now a W3C recommendation [22]. An interesting feature of OWL is that it relies upon a family of languages known as Description Logics (DL) that provide an inference system based on a formal well founded semantics [2]. The basic components of DL are *concepts*, *roles (properties),* termed as TBox, and *individuals,* termed as ABox. Concepts describe the common properties of a collection of individuals and roles are binary relations between concepts. Furthermore, a number of language constructs, such as intersection, union and role quantification, can be used to define new concepts and roles. The main reasoning tasks are *classification* and *satisfiability*, *subsumption* and i*nstance checking*. Classification is the computation of a concept hierarchy based on subsumption, whereas instance checking verifies that an individual is an instance of a concept.

OWL currently has three sublanguages (sometimes also referred to as 'species') of increasingly expressive power that are: OWL Lite, that is the syntactically easiest version, that can define hierarchies and simple constraints; OWL DL that allows the maximum expressiveness while retaining computational completeness, corresponds to Description Logics. Finally, OWL Full allows for maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. In this paper we exploit OWL DL for the formalization of the semantic trajectories, background/domain knowledge and reasoning. Here, ontological representation of semantic trajectories is immersed in an application ontology expressing domain knowledge. Indeed, domain knowledge is composed both of geographical knowledge, where the game takes place (Amsterdam), and expert knowledge about the game itself. The geographical knowledge is derived from a urban ontology that describes the taxonomy of places where the players move through (e.g. buildings, squares, bridges, parks). Temporal knowledge is encoded as discretization of time interval in *short*, *medium* and *long*. Furthermore, expert knowledge is described by means of axioms. Each axiom is a combination of logical operators that implicitly describes a class of objects.

In the following, we are interested in finding bad behaviour of kids. Indeed, we show an example of axioms that allow characterizing possible malicious player movements in terms of domain knowledge.

We first define which are the stops that could be inferred as being part of the game (used to solve the riddles), respect to other stops that could be lunch break or other not-game activities.

From the expert knowledge, we know that a **GAME_STOP** is a stop that is inferred to be related to a riddle solving, and has typically a medium duration. On the contrary, a stop not connected to the game has a long or short duration. Examples of not game stops are the lunch breaks and stops over the bridges launching stones in the water. Therefore, a possible characterization of a *malicious behaviour* of a kid is a trajectory with a *not game stop* that has duration *short* (`hasDuration`) and is located (`is_at`) *over a bridge*.

**NOT_GAME_STOP** ≡ `(StopHasTime some (hasDuration some LONG)) or (StopHasTime some (hasDuration some SHORT))`

**GAME_STOP** ≡ `StopHasTime some (hasDuration some MEDIUM)`

**MaliciousTrajectory** ≡ `trajIsCompOfStop some (NOT_GAME_STOP and (StopHasTime some (hasDuration some SHORT)) and (is_at some BridgePlace))`

This example, despite its simplicity, shows how OWL axioms allow performing automatic reasoning over semantic trajectories taking into account additional background information, such as geographical and application-dependent knowledge. As a result, the reasoner classifies semantic trajectories into people behaviour. This semantic enrichment process allows a user to query the ontology asking, for example, which are malicious trajectories and therefore derive bad behaviour of kids. It is worth noticing that such query could not be asked in a standard GIS or a Moving Object Database storing raw trajectories, due to the lack of encoded (application dependent) semantics.

This ontology has been developed using Protégé ontology editor [19], integrated with Pellet reasoner [18]. In this experiment Stops and Moves have been defined by means of spatial and temporal operators on a DBMS where data are stored [17]. For example, a stop has been found as the place and the time where the object does not move (or its movement is below a given threshold) [1]. After the Stop and Moves tables have been populated, they have been imported into Protégé by means of the DataMaster plugin [6]. Similarly, places have been extracted from spatial data and imported into Protegè.

However, despite the fact that knowledge representation by means of a formal ontology brings a fundamental advantage of having ready-to-use inference system providing useful reasoning services for granted, reasoning on individuals (i.e. ABox reasoning) is not scalable in any of DL language flavours. Clearly, this limitation is the major obstacle for combining ontologies with very large databases such as spatial and spatio-temporal datasets. Actually, reasoning scalability is an important open issue investigated in DL research domain.

Recent researches [16, 5] have pointed out some solutions for this problem. Calvanese et al, in [5] proposed to maintain data inside database and create mappings from ontology terms to database queries, exploiting database when reasoning on the data. In this approach, an ontology query is translated into several database queries, which means that query answering becomes the basic reasoning services for the ontology. A direct consequence of this approach is to devise (1) an ontological language that include the main modelling features of conceptual data models (e.g. ER model), and (2) a query language for the ontology that include at least conjunctive queries (corresponding to the select-project-join fragment of SQL). Thus, from the result of (1), the expressive power of DL language is reduced in order to correspond to conceptual data model. However, reducing DL language expressiveness jeopardizes the representation of complex axioms, extremely important for trajectory application domain. For example, trajectory patterns in our ontology are defined by axioms that cannot be represented using DL-Lite language proposed by Calvanese et al.

Another approach to cope with ABox reasoning on large knowledge bases is to exploit Oracle Semantic Technologies [14]. Indeed, in its latest 11g release, Oracle provides OWL support for knowledge representation and reasoning, namely the OWLPRIME language. Currently, we are working on translating the movement ontology in OWLPRIME to experiment both the potentialities and the limitations of this technology.

## 5 Conclusions and Future Work

In this paper we presented a trajectory semantic enrichment process for the interpretation of people movement. Indeed, raw trajectories as collected by mobile devices do not embed any kind of information about the travelling object itself neither about a possible interpretation of its movement in an application context. In order to provide a user with some encoded semantics, we propose a methodology that starts from a definition of semantic trajectories as stop and moves, translates it into an ontological formalism and then enriches the trajectory ontology with an application ontology equipped with reasoning axioms. This allows getting the maximum semantics in term of encoded application knowledge.

Future work includes defining new notions of semantic trajectories and the integration of data mining algorithms with ontological description. Indeed, some concepts defined in the ontology could be computed by complex data mining algorithms. For instance, moves and stops can be computed differently according to application requirements, or large sets of people movements need to be summarized as movement patterns. To this end, an approach that permits linking concept definition with data mining tasks is needed.

Privacy is a big concern when dealing with people movements [7]. It could be interesting to extend the semantic enrichment process to explicitly deal with privacy issues.

## Acknowledgements

This work has been partially supported by GeoPKDD EU Project IST-6FP-014915, http://www.geopkdd.eu/.

## References

1. Alvares, L.O., Bogorny, V., Kuijpers, B., Macedo, J., Moelans, B., Vaisman, A.: A Model for Enriching Trajectories with Semantic Geographical Information. In: Proc. of the ACM 15th International Symposium on Advances in Geographic Information Systems (ACM-GIS 2007) (2007)
2. Behr, T., Almeida, V.T., Güting, R.H.: Representation of periodic moving objects in databases. In: Forteenth ACM International Symposium on Geographic Information Systems (ACM-GIS 2006), pp. 43–50 (2006)
3. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F.: The Description Logic Handbook: Theory, Implementation, Applications. Cambridge University Press, Cambridge (2003)
4. Caglioni, Rabino, Ontologies of urban systems Joint Congress of the European Regional Science Association (47th Congress) and ASRDLF (Association de Science Régionale de Langue Française, 44th Congress)
5. Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Poggi, A., Rosati, R.: Ontology-based database access. In: Proc. of the 15th Italian Conf. on Database Systems (SEBD 2007), pp. 324–331 (2007)
6. Data Master Plugin for Protégé, `http://protegewiki.stanford.edu/index.php/DataMaster`
7. GeoPKDD EU Project IST-6FP-014915, `http://www.geopkdd.eu/`
8. Gruber, T.R.: A translation approach to portable ontologies. Knowledge Acquisition 5(2), 199–220 (1993)
9. Gruber, T.R.: Ontology. In: Liuand, L., Tamer Özsu, M. (eds.) Entry in the Encyclopedia of Database Systems. Springer, Heidelberg (to appear, 2008)

10. Guarino, N.: Formal Ontology and Information Systems, Formal Ontology and Information Systems. In: Guarino, N. (ed.) Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS (1998)
11. Güting, R., Schneider, M.: Moving objects databases. Morgan Kaufmann, San Francisco (2005)
12. Khatri, V., Ram, S., Snodgrass, R.: Augmenting a conceptual model with geospatiotemporal annotations. IEEE Transactions on Knowledge and Data Engineering 16, 1324–1338 (2004)
13. Noyon, V., Claramunt, C., Devogele, T.: A Relative Representation of Trajectories in Geogaphical Spaces, GeoInformatica, `http://www.springerlink.com/content/t7t274016234/?p=48df544df30f448387199a7f0b0b13a7&pi=0`
14. Oracle Semantic Technologies, `http://www.oracle.com/technology/tech/semantic_technologies/index.html`
15. Parent, C., Spaccapietra, S., Zimanyi, E.: Conceptual Modeling for Traditional and Spatio-Temporal Applications – The MADS Approach. Springer, Heidelberg (2006)
16. Poggi, A., Lembo, D., Calvanese, D., De Giacomo, G., Lenzerini, M., Rosati, R.: Linking data to ontologies. Journal on Data Semantics X, 133–173 (2008)
17. Pelekis, N., Theodoridis, Y.: Boosting Location-based Services with a Moving Object Database Engine. In: Proc. of Int. ACM SIGMOD/PODS Workshop on Data Engineering for Wireless and Mobile Access, pp. 3–10 (2006)
18. Reasoner, P.: `http://pellet.owldl.com/`
19. Protégé-OWL (ed.): `http://protege.stanford.edu/overview/protege-owl.html`
20. Spaccapietra, S., Parent, C., Damiani, M.L., de Macedo, J.A., Porto, F., Vangenot, C.: A conceptual view on trajectories. Data & Knowledge Engineering 65(1) (April 2008)
21. Teller, J., Lee, J., Roussey, C.: Ontologies for Urban Development. Studies in Computational Intelligence 61 (2007)
22. W3C Consortium. The web ontology language, `http://www.w3.org/TR/owlfeatures/`

# Administrative Units, an Ontological Perspective

Francisco J. López-Pellicer, Aneta J. Florczyk, Javier Lacasta,
Francisco Javier Zarazaga-Soria, and Pedro R. Muro-Medrano

Department of Computer Science and Systems Engineering,
Centro Politécnico Superior, University of Saragossa,
María de Luna 1, 50018, Saragossa, Spain
{fjlopez,florczyk,jlacasta,javy,prmuro}@unizar.es
http://iaaa.cps.unizar.es/

**Abstract.** The administrative units have been created with the purpose of covering specific territorial and functional scopes over time. Therefore, there are heterogeneity not only among states but also at any level of subdivision. In the context of Spatial Data Infrastructures, administrative units are part of the core data model and they are often exploited in the development of web services. International, cross-border, even national web services may face different and superposed administrative models. The administrative models are complex and they may not be well understood by users and developers in some scenarios, i.e. a query in boundary areas with different administrative models. This paper proposes an ontology that can describe administrative models and also serve as a knowledge base that may facilitate mappings between different types of administrative units.

**Keywords:** SDI, Ontology, Interoperability, Administrative Model.

## 1 Introduction

Spatial Data Infrastructures (SDI) are a coordinated approach to technology, policies, standards, and human resources necessary for the effective acquisition, management, distribution and utilization of geographic information at different organization levels and involving both public and private institutions. This effort has resulted in the formation of cross-jurisdictional partnerships as is stated in Rajabifard et al. [15]. Cross-jurisdictional partnerships often implies services and data models able to deal with different kinds of administrative. For example, the European Union (EU) directive establishing an Infrastructure for Spatial Information in the European Community (INSPIRE), that promotes the development of SDIs in Europe, includes the administrative units as one of the spatial themes that should be harmonized first.

Modern life requires clearly bounded territorial spaces which act as geographical containers of social processes. Territorial space means here a social constructed place (e.g. Spain) generically dependent on a physical place (e.g. a region on Earth bounded by coordinates). Along with the territorial space, the society construct special social organizations for the governance of portions the

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 354–363, 2008.

territorial space over time. These entities are units of administration for local, regional, national or international governance with specific roles separated by crisp administrative boundaries. In this paper we use the term *jurisdictional domain* (JD) to identify them. This term comes from the context of business transactions [9]. A jurisdictional domain is a territorial jurisdiction that is source of legal constraints for rational agents (e.g. an human being, an organization) and other jurisdictional domains, often dependent (e.g. a county). Each jurisdictional domain controls a geographical extent that governs and can create other jurisdictional domains within the extent of its jurisdiction. These new jurisdictional domains are recognized by law as distinct legal and/or regulatory frameworks. For example, in order to ease the territorial management, states often allows administrative divisions (e.g. provinces, territories, cantons, länders, etc.) to create their own administrative subdivisions. Jurisdictional domains can also be combined to form new entities with an associated extent as big as a continent (e.g. EU) or as small as a river island (e.g., Pheasant Island, condominium of Spain and France located in the River Bidasoa).

Jurisdictional domains are not static entities. They are created, destroyed or merged. Their properties may also vary: their associated extension can be modified, and even they can be transformed into another type of entity. In the same way, the original purpose of the entity can evolve along time.

Administrative units are far from being adjusted to a stable and uniform hierarchy of types and instances. The complexity in their diversity and peculiarities mixed with its evolving nature has created the necessity to provide a coherent model that might simplify their use in SDI systems. This paper proposes a representation of administrative units based on a reusable domain ontology, which defines the general structure of the units and their relationships. Additionally the paper provides an example of application ontology describing the administrative units system of Spain.

The paper is organized as follows. Section 2 presents the state-of-art. Section 3 describes the domain ontology and its characteristics. Section 4 presents the application ontology. Section 5 shows the uses of the ontology in an SDI. The paper ends with conclusions and further work.

## 2 State-of-Art

In the SDI context there are works such as Irie et al. [10], Manov et al. [12] and international standards such as ISO 19109:2005 *Geographic information - Rules for application schema* (ISO 19109) that propose general purpose models able to represent any type of geographical entity. In the narrower scope of the administrative units, there are administrative schemes based on different knowledge organization models such as lists [14, 16], thesauri [6] or ontologies [4] to describe the structure of the different countries. They use the generic definition of feature as "a meaningful object in the selected domain of discourse" (ISO 19109) and support geographic relation types. Others, as the international standard ISO 19112:2003 *Geographic information - Spatial referencing by* *geographic identifiers*, describe the logical model of an authorized dictionary of names (gazetteer) and present the administrative units as a hierarchy.

However, as some experts suggest in Bleakly [1], they do not consider important issues such as the unique identification of items (unique name ID), multilingualism, duration (time frame for names), reliability of data (source reliability, data accuracy), spatial characteristics (elevation, map and image files, and both, point and bounding box for coordinates) and tabular data (population data); moreover, the most common characteristic of the above models and schemes is the lack of an appropriate semantic representation of the types of administrative units and their spatial and temporal relations.

## 3 Ontology of Administrative Units

### 3.1 Framework

Our proposal is the result of the analysis of three existing standard models: the Nomenclature of Territorial Units for Statistics (NUTS) developed by the EU; the FIPS 10-4 standard for countries, dependencies, areas of special sovereignty and their principal administrative divisions developed by the United States Federal Government; and the ISO 3166 Codes for the representation of names of countries and their subdivisions.

It has been possible to identify the common elements used for referencing real or instrumental countries, dependent areas, and subdivisions with a political, statistical, environmental or commercial purpose. This analysis has detected among others problems that the provided set of units might not be exhaustive (lack of some subdivisions of the administrative units), there is no guarantee that the name used to identify the unit is administratively recognized and there is no consistency in the representation of the spatial properties.

Furthermore, the most difficult problem detected is that these models follow the vernacular hierarchical view based on the perception of the administrative unit as a geographic container. To deal with this issue we propose an approach based on the development of dual geographic and administrative hierarchies.

We follow the scheme proposed by Guarino [8] for building domain and application ontologies. This scheme has three layers:

- (1) A high-level ontology that defines data types and general relations which are independent of context.
- (2) A domain ontology which defines concepts and relations that can be reused in the context of the administrative models of different countries.
- (3) And an application ontology per country, which represents the specific types of administrative units of each country, along with specific instances of existing units.

As high-level ontology, DOLCE [5] has been selected because it contains all the basic concepts and relationships needed to build the domain ontology. DOLCE is $\mathcal{SHION}(\mathcal{D})$ in description logic. OWL-DL closely correspond to it with some limitations on datatypes. The use of this high-level ontology in other environments simplifies the combination of our domain ontology with existing ones. The predicates previously presented in DOLCE literature [2, 3, 11, 13] we will refer to are:

- $\mathtt{ORG}\,(\mathtt{x})$ standing for "x is an organization", a socially-constructed person with a complex articulation of tasks, roles and figures that has sovereignty over a definite territory.
- $\mathtt{PGO}\,(\mathtt{x})$ standing for "x is a political geographic object", i.e. a geographical place, conventionally accepted by a community.
- $\mathtt{COL}\,(\mathtt{x})$ standing for "x is a collection", i.e. a federation is collection of states.
- $\mathtt{INST}\,(\mathtt{x},\mathtt{y})$ standing for "x institutionalizes y" when such a concept "x" is used by a description that is valid for "y", i.e. public administration can be applied as a province description.
- $\mathtt{PRE}\,(\mathtt{x},\mathtt{t})$ standing for "x is present at time t", i.e. France is present now.
- $\mathtt{MEM}\,(\mathtt{x},\mathtt{c},\mathtt{t})$ standing for "x is member of c at time t", that implies "c" is a collection by definition, i.e. Spain is member of the EU.
- $\mathtt{PC}\,(\mathtt{x},\mathtt{y},\mathtt{t})$ standing for "x is part of y at time t"
- $\mathtt{GP}\,(\mathtt{x},\mathtt{y},\mathtt{t})$ standing for "x is geographic part of y at time t", that implies "x" and "y" are political geographic objects by definition.

### 3.2 Domain Ontology

The next step is of defining what a jurisdictional domain is and its basic taxonomy (see Fig. 1). To deal with the hierarchical view based on the perception of the administrative unit as a geographic container we need to define a concept that only holds spatial information. We introduce the concept of *jurisdictional geographic object* (JGO). It is the spatial area on which a *jurisdictional domain* rules and depends on. Here it is sufficient to point that jurisdictional geographic objects are political geographic objects whose spatial properties may vary over time.

$$\mathtt{JGO}(\mathtt{x}) \rightarrow \mathtt{PGO}(\mathtt{x}) \tag{1}$$

A *jursidictional domain* (JD) is defined as any social entity recognized by the law as a distinct legal and/or regulatory framework with the role of public administration. Jurisdictional domains are organizations which are described by the role *public administration* and are grounded by dependant jurisdictional geographic objects during the whole period in which the jurisdictional domain is present:

$$\begin{aligned} \mathtt{JD}\,(\mathtt{x}) \rightarrow\ & \mathtt{ORG}\,(\mathtt{x}) \wedge \mathtt{INST}\,(\mathit{PublicAdministration},\mathtt{x}) \\ & \wedge \exists \mathtt{t}\,(\mathtt{PRE}\,(\mathtt{x},\mathtt{t})) \wedge \forall \mathtt{t}\,(\mathtt{PRE}\,(\mathtt{x},\mathtt{t}) \rightarrow \exists \mathtt{y}\,(\mathtt{JGO}\,(\mathtt{y}) \wedge \mathtt{PC}\,(\mathtt{y},\mathtt{x},\mathtt{t}))) \end{aligned} \tag{2}$$

The jurisdictional domain may be described playing other roles which are defined upon the functions that the administrative unit may have. For example, "local power" is the role of municipality (the closest to citizens). The jurisdictional domain concept may be specialized in states, administrative divisions and

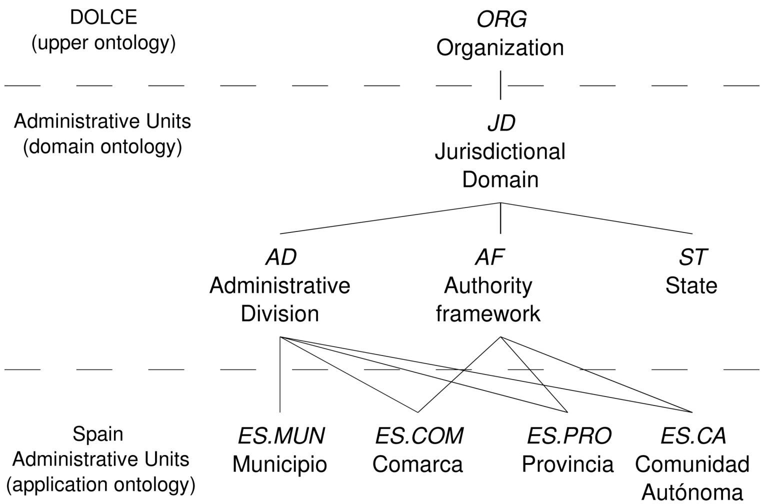



**Fig. 1.** The jurisdictional domain taxonomy: main concepts and a subset of concepts drawn from an application ontology

authority frameworks. An *administrative division* (AD) represents any division in a jurisdictional domain. This concept is characterized by the temporal part-hood relation which relates with their parent jurisdiction. Examples of instances are Saragossa, Huesca and Teruel which are Municipalities of Spain.

$$\mathtt{AD}(\mathtt{x}) \rightarrow \mathtt{JD}(\mathtt{x}) \wedge \forall \mathtt{t}\,(\mathtt{PRE}\,(\mathtt{x},\mathtt{t}) \rightarrow \exists \mathtt{y}\,(\mathtt{JD}(\mathtt{y}) \wedge \mathtt{PC}\,(\mathtt{x},\mathtt{y},\mathtt{t}))) \tag{3}$$

An *authority framework* (AF) represents any jurisdictional domain constructed as aggregation of other jurisdictional domains. Jurisdictional Domains have no restriction in the number of memberships. Examples of instances are the EU (an aggregation made of States) and the Warsaw voivodeship in Poland (an aggregation made of counties), which is also an administrative division.

$$\mathtt{AF}(\mathtt{x}) \rightarrow \mathtt{JD}(\mathtt{x}) \wedge \forall \mathtt{t}\,(\mathtt{PRE}\,(\mathtt{x},\mathtt{t}) \rightarrow \exists \mathtt{y}\,(\mathtt{JD}(\mathtt{y}) \wedge \mathtt{MEM}\,(\mathtt{y},\mathtt{x},\mathtt{t}))) \tag{4}$$

A *state* (ST) consist of a bordered territory under effective and civil government. In Max Weber's words, [17] have the "monopoly on the legitimate use of physical force within a given territory". This concept disjoint administrative division. State instances are often the root element in many administrative code lists (e.g., FIPS 10-4, ISO 3166, NUTS). Examples of instances are the French Republic, United Kingdom and Spain.

$$\mathtt{ST}(\mathtt{x}) \rightarrow \mathtt{JD}(\mathtt{x}) \wedge \neg\mathtt{AD}\,(\mathtt{x}) \tag{5}$$

### 3.3 Application Ontology

The administrative unit model of Spain is quite complex. Table 1 shows the most important types. Territorial areas separated from the mainland have their own

**Table 1.** Spain administrative units[14]

| Name | Description | Units | ADL Feature Type |
|---|---|---|---|
| Comunidad Autónoma | autononomous community | 17 | countries, 1st order division |
| Ciudad Autónoma | autonomous city | 2 | |
| Provincia | province | 50 | countries, 2nd order division |
| Isla | island | 11 | |
| Veguería | group of district (Catalonia) | 6 | |
| Comarca | district (Aragon, Basque Country, Castile and Leon, Catalonia) | 81 | countries, 3rd order division |
| Mancomunidad | municipality association | 1.019 | |
| Área metropolitana | metropolitan area | 4 | |
| Facería | common land (Navarre) | 64 | |
| Municipio | municipality | 8.111 | |
| E.Á.T.I.M. | minor civil unit | 1.019 | countries, 4th order division |
| Parroquia | parish (Galicia) | 3.781 | |

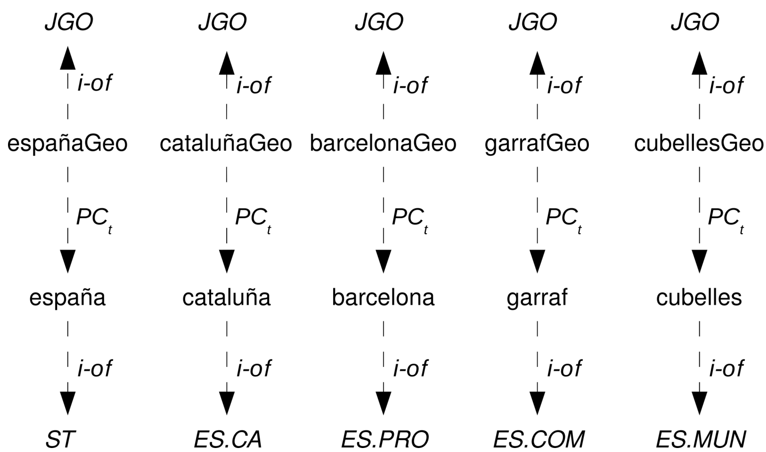



**Fig. 2.** Elements of the application ontology

special administrative units (Isla, Ciudad Autónoma). Some autonomous communities have their own administrative subdivisions (Aragon, Basque Country, Castile and Leon, Catalonia, Galicia and Navarre). Fig. 2 shows only a subset of the application ontology in the scope of the autonomous community of Catalonia. The following conventions are assumed: concepts are represented in capital letters, individuals are represented in small letters, relations between individuals are represented by dashed labeled arrows and the relation between an individual and the concept is labeled by *i-of*. The jurisdictional domains shown here are state (ST), autonomous community (ES.CA), province (ES.PRO), district (ES.COM) and municipality (ES.MUN). Each jurisdictional domain is related with their jurisdictional geographic object which represents the physical area where the unit governs. Fig. 3 shows the geographic containment among the


**Fig. 3.** The geographic containment hierarchy

jurisdictional geographic objects. This is the typical geographic containment hierarchy that can be found in models such as Geonames [7]. Our application ontology can model the more complex relations among jurisdictional objects as figure 4 shows. This figure shows that Catalonia, Barcelona and Cubelles are part of Spain. Garraf is also part of Catalonia because is a subdivision of Catalonia and because the part relation is transitive is also part of Spain. Cubelles is also part of Catalonia as constituent. Barcelona (province) and Garraf (district) are defined by law as an aggregation of municipalities. Why Garraf is not part of Barcelona? Why Barcelona is not part of Catalonia? Because they belong to different but spatially superposed administrative hierarchies.

## 4 Applications of the Administrative Unit Ontology

The administrative relations added by the ontology improve the conceptual search. The addition of spatial restrictions allows the construction of more powerful queries. Furthermore, the ontology can facilitate the identification of equivalent units from different administrative structures thanks to the alignment of roles, their spatial characteristics and their position in the hierarchy. The proper management of different administrative organization models is crucial for the behaviour of SDI services and service chaining functionality. Assuming that two administrative units are equivalent when they play equivalent roles, one can create a complete map of roles shared among units of different administrative unit models. This approach could facilitate the management of resources in border areas. For example, let us think about the search for local ski facilities in municipalities of the Pyrenees. Spanish users may ask for "Municipios" and queries are made about "Muncipios" and "Communes" because we may have inferred that

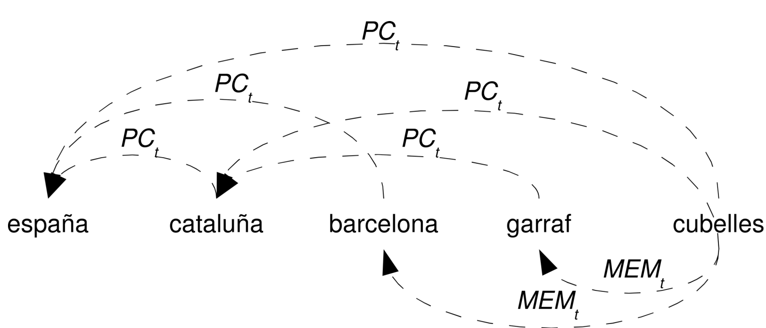



**Fig. 4.** The multipath administrative hierarchy

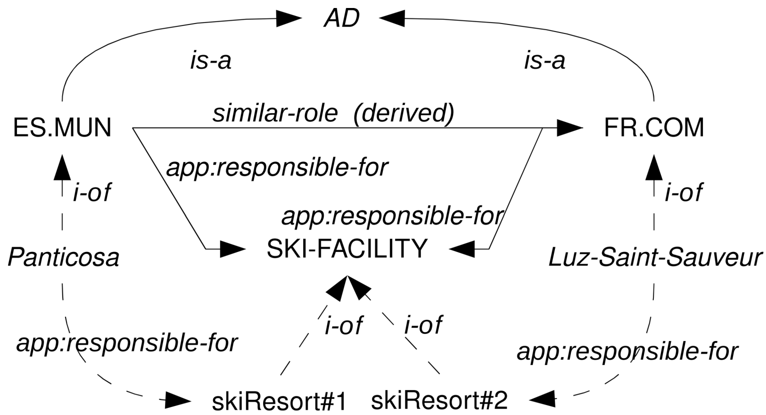



**Fig. 5.** Municipalities and ski resorts

they play a similar role (Fig. 4) as they have a similar position in their respective hierarchies and are responsible of ski facilities.

## 5 Conclusions and Future Work

This paper has presented an administrative unit ontology to model the administrative structure of a country and it has described possible advantages derived from the use of such ontology.

The following steps of this work will be the development of a semiautomatic process to generate the administrative instances for the cases of France, United Kingdom, Portugal and Spain. This task can be more complex than expected because of the complexity and diversity in the administrative structure of each country, and the difficulty of obtaining official data. For example, in Spain, the Ministry for Public Administration has a registry of local administrative units, but each autonomous community may have its own registry for their specific administrative subdivisions. Therefore, a whole model always needs to merge data form quite different sources.

It is important to stress that the high frequency of changes in the administrative organization (shape, structure, name or administrative capabilities) makes necessary to establish specialised policies and techniques for updating all the elements of the infrastructure that uses this model. Also the quality of the data sources is an issue that should be considered. If the official name, code or coordinates of a unit are not accurate, associated services in the SDI that use this information will obtain poor results.

With the resulting administrative knowledge base, the efforts will focus on providing mechanisms for service chaining and semantic annotation for SDI based on the knowledge base.

## Acknowledgments

The basic technology of this work has been partially supported by the Spanish Ministry of Science and Technology through the project TIN2007-65341 from the National Plan for Scientific Research, Development and Technology Innovation, by the University of Saragossa through the project UZ200700024 and by GeoSpatiumLab S.L

## References

[1] Bleakly, D.R.: Final report: PATTON Alliance gazetteer evaluation project. Technical Report SAND2007-5260, Sandia National Laboratories (2007)
[2] Bottazzi, E., Ferrario, R.: Preliminaries to a DOLCE Ontology of Organizations. International Journal of Business Process Integration and Management, Special Issue on Vocabularies, Ontologies and Business Rules for Enterprise Modeling (2008)
[3] Bottazzi, E., Catenacci, C., Gangemi, A., Lehmann, J.: From collective intentionality to intentional collectives: An ontological perspective. Cognitive Systems Research 7(2-3), 192–208 (2006)
[4] Chaves, M.S., Silva, M.J., Martins, B.: A Geographic Knowledge Base for Semantic Web Applications. In: Heuser, C.A. (ed.) Proceedings of the 20th Brazilian Symposium on Databases, October, 3–7 2005, pp. 40–54. Uberlândia, Minas Gerais (2005)
[5] Claudio, M., Nicola, G., Alessandro, O., Luc, S.: The WonderWeb Library of Foundational Ontologies. Wonderweb deliverable d18, Laboratory For Applied Ontology - ISTC-CNR (2005)
[6] European Communities and Spain Senate. EUROVOC. This version includes the municipalities of Spain (last accessed June 21, 2008), http://www.senado.es/tesauro/intro.html
[7] GeoNames. GeoTree (last accessed June 21, 2008), http://geotree.geonames.org
[8] Guarino, N.: Formal Ontology and Information Systems. In: Guarino, N. (ed.) Proceedings of the First International Conference on Formal Ontologies in Information Systems (FOIS 1998), Trento, Italy, June 6-8, 1998, pp. 3–15. IOS Press, Amsterdam (1998)
[9] International Organization for Standardization (ISO). ISO 15944-5: Information technology – Business Operational View – Part 5: Identification and referencing of requirements of jurisdictional domains as sources of external constraints. Technical Report ISO 15944-5, International Organization for Standardization (ISO) (2008)
[10] Irie, R., Sundheim, B.: Resources for Place Name Analysis. In: Fourth International Conference on Language Resources and Evaluation - LREC 2004, May 24-30, 2004, pp. 317–320 (2004)
[11] Laboratory for Applied Ontology, Institute of Cognitive Sciences and Technology of the Italian National Research Council. Dolce-2.1-lite-plus (2006) (accessed June 21, 2008), http://www.loa-cnr.it/ontologies/DLP_397.owl

[12] Manov, D., Kiryakov, A., Popov, B., Bontcheva, K., Maynard, D., Cunningham, H.: Experiments with geographic knowledge for information extraction. In: Kornai, A., Sundheim, B. (eds.) HLT-NAACL 2003 Workshop: Analysis of Geographic References, Edmonton, Alberta, Canada, May 31 2003, pp. 1–9. Association for Computational Linguistics (2003)
[13] Guarino,: Social Roles and their Descriptions. In: Dubois, D., Welty, C.A., Williams, M.-A. (eds.) Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR 2004), Whistler, Canada, pp. 267–277. AAAI Press, Menlo Park (2004)
[14] Ministry for Public Administration of Spain. Spanish Registry of Local Entities (accessed June 21, 2008), http://www.map.es/documentacion/entes_locales/registro_entidades_locales.html
[15] Rajabifard, A., Feeney, M.-E.F., Williamson, I.P.: Future directions for SDI development. International Journal of Applied Earth Observation and Geoinformation 4(1), 11–22 (2002)
[16] Spanish National Institute of Statistics. List of place names (accessed June 21, 2008), http://www.ine.es/nomen2/index.do
[17] Weber, M.: Max Weber: Essays in Sociology, pp. 77–128. Oxford University Press, New York (1946); Originally a speech at Munich University (1918) (accessed June 21, 2008), http://en.wikisource.org/wiki/Politics_as_a_Vocation

# A Modular Data Infrastructure for Location-Based Services

Shijun Yu and Stefano Spaccapietra

Database Laboratory, I&C, EPFL, Switzerland
shijun.yu@epfl.ch, stefano.spaccapietra@epfl.ch

**Abstract.** Knowledgable Location-based Services (LBS) aim at enabling mobile users to specify their requests and profiles on the move and providing them with context-aware and personalized local information relevant to their current activity and request. Therefore, it opens up new challenges in data infrastructure, knowledge representations and data management etc. In this paper, firstly we will discuss the characteristics of LBS in terms of data management, and then present our data architecture. Finally, we will explain how the knowledge is incrementally set up and maintained in a modular manner.

## 1 Introduction and Motivation

Traditional data management applications operate in well-structured information environments and benefit from full-fledged strategies (e.g. SQL) that provide the needed functionality to represent, manage and query well-structured data. Web-based application environments, on the other hand, face a cumbersome task trying to provide the same functionality for the huge amount of heterogeneous data resources that reside on the Web. This new framework entails an emphasis on elicitation of data semantics, to improve the chances for correct interpretation of the data by heterogeneous partners (users and agents) that do not adhere a priori to common coordinated behavior. In parallel, users' increased mobility has lead to the development of Location-Based Services (LBS), i.e. services tailored to provide information that is selected taking into account the current location of the user on the move. Current LBS rely on traditional data management techniques, integrating all the data they need into a single centralized repository and providing all their users with the same services. We foresee that a new generation of LBS, which we call knowledgeable LBS, will be developed to provide enhanced services, namely contextualized and personalized information retrieval, and constant evolution to acquire new knowledge as available and as requested by users. Given the commonality between knowledgeable LBS and semantic web services in terms of facing an unbounded information space, LBS will likely use semantic web technology, enabling them to be used not only by users on the move, but also via the web, and exchange data and services with any agent within the semantic web. LBS services will nevertheless retain their specificities, namely:

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 364–374, 2008.

- **Locality, Mobility and Dynamics.** LBS web services will provide information about a specific region, in particular information related with the current real or virtual location of the user. To fulfill this capability, LBS have to build and maintain knowledge repositories describing all locally relevant data stored in a limited number of data sources. When the user moves from her/his current place to a remote one (e.g. from Paris to London), the previous sources and knowledge become useless for upcoming queries from the user. Therefore, LBS need the capacity to dynamically acquire new sources and build the corresponding new knowledge into its repositories. Alternatively, LBS can be specialized to only serve a specific region. In this case, it will be up to users to switch from one LBS to another (just like cellphone users today switch from one telecommunication provider to another one). This paper gives some hints on how LBS may become knowledgeable about a specific region, which we see as a basic functionality for 2nd generation LBS.
- **Trading comprehensiveness for rapidity.** LBS intend to serve people on the move, i.e. they have to provide rapid rather than comprehensive responses to user queries. Well-known centralized management strategies that call for long set up processes and static solutions are not well suited to LBS needs. This departs significantly from e.g. data warehousing frameworks [4] that have similar data heterogeneity problems but different data quality and comprehensiveness requirements. Suitable query processing strategies need to be developed.
- **Modularity.** LBS obviously have a strong specific focus on spatial and temporal information and related constraints. In addition to space and time challenges, LBS aiming at context-awareness have to be highly sensitive to the current state of affairs when responding to user requests. Similarly, they have to care about personalizing services based on knowledge they can acquire about user's characteristics. This multiplicity and diversity of concerns make LBS a complex software whose processes constantly need to adjust to running circumstances. To make this possible while keeping performance we propose a modular data architecture, more easily manageable than the centralized approach in current LBS.

## 2 Related Work

Many different strategies can be applied in a LBS to organize, acquire and maintain the heterogeneous data involved in LBS. Earlier LBS were mainly concerned about a specific application, for instance, Active-Badge [9] to obtain the latest target's location, EasyLiving [7] to *'find a colleague in the building'*. In these systems, LBS infrastructures were relatively simple and mostly oriented towards to a single application supported by a central database. From a practitioners' viewpoint, a middleware-based architecture [1] is well-suited for LBS design due to its application-independence, adaptability to multiple platforms and dynamicity. A positioning service, for example, is an obvious component of a LBS middleware model [3]. Similar proposals are also embodied in [2] and commercial products at IBM, ORACLE and Microsoft. In CRUMPET project [6], client-mediator-server three-tier approach was employed to improve the context-aware interactions between the user and services.

Crucial components for knowledgeable LBS are those supporting semantic interoperability and data reusability. As we have learnt from semantic web research, knowledge sharing and reasoning capabilities call for ontology management services. LBS need an ontological approach to solve typical issues about concept classification, knowledge inference, and concept similarity evaluation, just to name a few. We therefore propose a knowledge infrastructure relying on a domain ontology, alike e.g. a tourism ontology. In line with a web services view, we see the LBS ontology as an ontology of services and service usability. Because of this service orientation, the LBS ontology is to be equipped with specific features, such as links between services to show functional equivalence used to plan alternative services. Beyond service descriptions, our LBS ontology includes relevant contextual information that is essential to refine service usability given the current state of the local world. Similarly, it includes knowledge about user characterization used for personalization purposes. This makes up three knowledge modules that together define what we call the modular core ontology [5] as shown in Figure 1. Different from most ontologies, the LBS ontology has to have the generic knowledge and know-how to deal with space and time aspects and reasoning. For example, it has to know about points, lines and areas and about topological constraints. To emphasize this, Figure 1 shows two knowledge containers about space and time included in the core ontology. However, they are drawn differently because of the difference in nature wrt the user, context and service modules. Notice that Figure 1 only shows the repositories in the infrastructure, not the corresponding process components (cf. Section 6) used to build and maintain the repositories. Complementing the core ontology with rich terminological knowledge is important to support the variety of cultural and linguistic habits of LBS partners, through flexibility of vocabulary in the exchanges between the LBS and its partners (users, service providers, knowledge providers). The framework for these exchanges is provided by the mappings (cf. Figure 1) between LBS knowledge and the specific descriptions of the current LBS partners. We denote the latter as user/service/context profiles.

This paper presents and discusses the knowledge architecture that organizes the various components mentioned above, and how this knowledge is incrementally set up and maintained.

## 3 Our LBS's Data Infrastructure

By definition, a LBS holds an integrated view of data and services from sources local to a specific region (e.g. a city and its suburbs). Occasional extension of the geographic coverage can be achieved by considering the LBS as a peer in a peer-to-peer architecture [10]. Whenever the LBS is unable to answer a user query because the local data does not lead to a possible answer, before replying negatively it may forward the query to geographically neighboring LBSs. For instance, let us assume that user Shirley, currently in Lausanne, has to attend a meeting in Geneva later in the day. She asks for time-tables of relevant buses in Geneva. The LBS serving Lausanne, aware it is unable to answer her query, may forward the query to the neighboring LBS in Geneva. This paper does not address such typical P2P issues, it focuses instead on semantic data management for a single LBS.

Regarding data management for a single LBS, we do not propose to integrate or align all data from the sources into a single repository. Instead, LBS knowledge only includes an abstract view of the source data. Thus, for example, the detailed description of a service as stated by the service provider remains at the source, while the abstract view maintained in the LBS only records the main characteristics of the service, i.e. the minimal information that is needed to quickly estimate if, given a specific query, the service may be relevant or not for the querying user. Whenever more detailed information about a service is needed, the sources are directly queried to extract the instantiations and other properties of services that may be of interest to the requesting user. This approach shows two benefits: firstly, sources can autonomously maintain their data, while the LBS is just responsible for suggesting users where to find appropriate services. Secondly, the sources can protect their data based on their own privacy regulations and can constrain users' access so as to only provide their data if certain conditions are satisfied.

In order to build a knowledgeable LBS system capable of actually relating concepts from different sources, disambiguating the terms in queries, and supporting

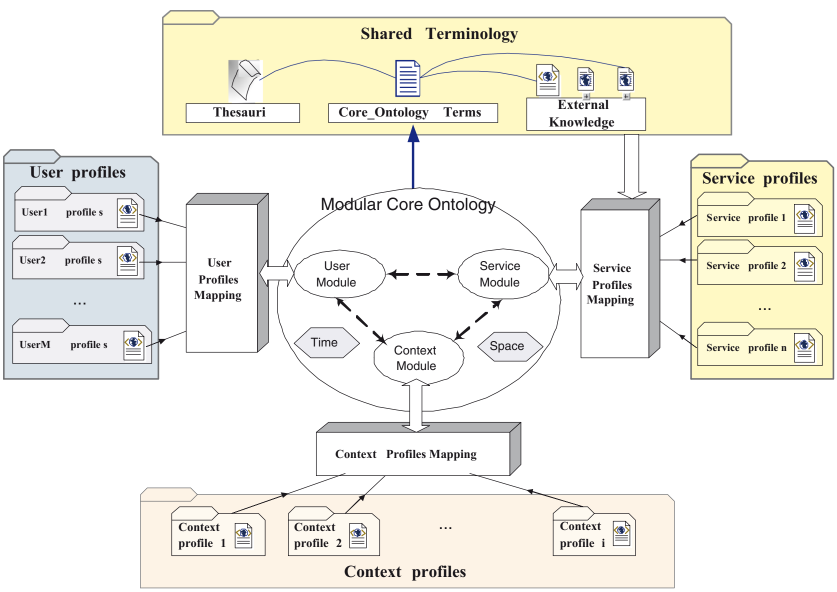



**Fig. 1.** The Basic Data Infrastructure in our LBS

query-service matching according to multiple criteria, so as to efficiently respond to user queries, the data infrastructure we propose includes six repositories interconnected via corresponding mappings (see Fig. 1). Two repositories, called Core Ontology and Shared Terminology, contain the domain-relevant knowledge built by the LBS and LBS administrators, and its related terms. Three other repositories contain knowledge on the external actors with whom the LBS interacts, i.e. users, services, and context providers. We say these repositories respectively contain the user profiles, the service profiles and the context profiles. Finally, the last repository (not shown in Fig. 1) is a working repository that contains the queries being processed by the LBS, thus holding data specific to a given query, and possibly the historical record of this data. For term similarity, we refer to this query data as query profiles. The infrastructure components are briefly described hereinafter to give an overall but intelligible view of the proposed infrastructure.

## 4 The Modular Core Ontology

The core ontology (CO) is the repository for the semantics of the data managed by the LBS. It is the kernel of the LBS's data infrastructure and it is used to structure the diverse aspects of service-related information and perform reasoning about it. It basically describes taxonomies of interest, encompassing a set of definitions of classes, properties, relations, axioms and constraints. These taxonomies organize information in complementary sub-domains. Services, users, context, space and time are the main sub-domain we have identified as essential to LBS. These sub-domains are quite heterogeneous and each one can be pretty complex in itself. Therefore, for better management and improved performance, we propose that the core ontology be a modular ontology composed by one module per sub-domain. A module is defined as a smaller ontology, which covers a sub-domain within the domain of the larger ontology. This fits perfectly with the LBS core ontology and its multiple taxonomies. Moreover, building modular ontologies is nowadays feasible. Several proposals exist on how to build a modular ontology, how to maintain modules individually as well as maintain the inter-module links that allow interactions between modules, and how to perform distributed reasoning within a modular organization [5]. Modular ontologies have been claimed to solve scalability issues and speed up reasoning, benefits that are for sure also relevant for LBS. However, the primary benefit we see in a modular ontology is its improved understandability and easiness of design and administration. A modular organization allows autonomous development of each module by an administrator with specific expertise in the sub-domain. Moreover, considering the semantic web and its world of specialized services, the development of modular approaches can improve the ontology's reusability. If good sub-domain repositories are available, it is much easier to reuse them than to elaborate a brand new one from scratch.

Whether modular or not, reuse is anyway the approach to follow when starting building the core ontology. Initializing the core ontology means inserting all concepts that are assumed to be useful for the targeted application(s). Once the targeted knowledge domain identified, a clever designer looks for existing ontologies in the same or similar domain. If any one is found, its import can form the initial set-up for the core ontology. For example, if a tourist-support application is targeted, the designer will look for, and find, a tourism ontology whose concepts include accommodation, food, transport, and leisure services. Very likely, many of the imported concepts will be generic enough to be suitable for the new LBS and its service module. However, not all of them will be relevant for local use, and not all of them will be formulated in a way that is consistent with local habits. Therefore, in a second step, the core ontology is turned into a *local* domain ontology (e.g. tourism support in the Canton de Vaud, Switzerland). This can be done by acquiring and adding location-specific information (i.e. contextual data), such as local landmarks and local calendars, and enriching existing information, such as adding the preconditions for using a given available service, e.g. to use a motor-boat rental service the user needs to hold a valid sailing licence. Conversely, making the ontology local also includes removing generic concepts that are locally irrelevant (e.g. downhill skiing for an LBS about Amsterdam).

At this point, the LBS is fully ready to start operation. As long as the LBS is in use, the core ontology is expanded based on the queries received and answers given. Identification of new requirements by the ontology manager will also lead to ontology expansion, but this is very much similar to normal ontology evolution, not specific to LBS, and will not be discussed here. Notice that for the administration of the core ontology given this incremental strategy it is advisable that elements in the ontology be qualified as either prospective or confirmed. A prospective element is one that has been entered in the initialization phase but has not yet been used (up to now). A confirmed element is one that has been actually used during the processing of at least one query. Prospective elements should sooner or later become confirmed elements. Elements that remain prospective elements for too long are candidate for deletion, to be triggered by the ontology managers whenever it is felt that this is a reasonable enhancement to ontology performance (smaller ontologies may be explored and updated faster than large ontologies). More sophisticated maintenance strategies may be defined, based on usage metrics, relevance feedback and other usability criteria. They are not investigated here.

## 5 The Shared Terminology

In our proposal, the set of terms hold by the shared terminology is a superset of the terms in the core ontology. This superset is intended to provide extended terminological support to facilitate interoperability among components of the data infrastructure. It helps in solving heterogeneity issues (from the syntactic level to semantic level) such as differences in the vocabulary of data sources, core ontology and user queries, and related ambiguities. A typical example is the use of synonyms, e.g. a user querying for soccer matches while the ontology records information on football matches. The shared terminology is queried by the LBS whenever the core ontology cannot identify a concept used in a query. For instance, a multi-lingual shared terminology may be used to overcome language limitations of a LBS based on a monolingual core ontology. An English-based LBS may thus be able to answer queries from French-speaking user.

We propose to initially populate the shared terminology with the concepts in the core ontology, which we call *internal terms*. Next, each internal term is complemented with its definition(s) extracted from certain thesauri. Subsequently, the shared terminology is further enriched by introducing what we call *external terms*, i.e. terms (from the thesauri) that are somehow relevant to the internal terms, but are absent from the core ontology. *Relevant* means there is a relationship between the internal term and the imported external term. For instance, the internal term 'car rental' may be related to the external term 'hire a car' with a semantic equivalence (synonymy) relationship. During LBS operation any new term appearing in user queries or in service-profiles will be identified thanks to the shared terminology manager and added to it in order to capture the terminologies of users and services that differ from the terminology of the LBS designers. Because the shared terminology manager can use any external ontology to identify the unknown term, we do not expect the identification process to fail. Should this happen, the query or service description using the term cannot be accepted and further human interaction is needed to solve the issue.

## 6 Information Profiling and Semantics Matching

**Service Profiles.** In our proposal, the descriptions of specific services are autonomously created by service providers and are kept and maintained at the corresponding data sources, external to the LBS. These service descriptions, together with some metadata about the data source (e.g. owner name, last update date), form what we call a service profile (see detailed definitions in [11]). Service profiles are acquired by the LBS when the data source joins the LBS. The acquisition process includes

**Service Profile Matcher.** This process component, the SP matcher, is responsible for acquiring the service profiles, recognizing the terms and concepts in the service descriptions, evaluate their matching with the knowledge in the service module of the core ontology, and accordingly establish the mappings between service profiles and service module. This mapping is used at query time to retrieve the services relevant to users' queries. The matching process uses a set of pre-defined syntactic and semantic rules to transform the heterogeneous service profiles into the format consistent with the LBS core ontology. The matching also produces the mapping of the core ontology into service descriptions, i.e. the syntactic and semantic support needed to transform users' queries into the format that the data sources can understand. The process is repeated for the classes, properties, and other features that the service profile encodes. The goal is not to fully copy the service profile descriptions into the ontology, but to ensure that the core ontology service module holds enough information about the service to be able to evaluate the effectiveness of the service as a response to users' queries.

**User Profiles.** The LBS needs to know about its users to achieve its personalization goal. Descriptions of users, mostly in terms of ¡attribute, value¿ pairs about e.g. age, profession, preferences and dislikes, are traditionally called user profiles. We assume these profiles are stored in a dedicated repository that may be within the LBS or in a site elsewhere located but accessible by the LBS. Same as for service profiles, user profiles need to be understood by the LBS, i.e. the LBS must identify what the data in the user profile means and how it can be related to context and service data. The LBS user module maintains the concepts generically related to users, possibly abstracted from the actual user profiles.

**User Profiles Matcher.** This process component, the UP matcher, plays for user profiles management the same role and functionality the SP matcher plays for service profiles. A specificity of the UP matcher is to extract from the user profile information to guide the presentation of the query results to the user.

**Context Profiles.** The LBS needs to know what information participates into the description of the local framework (e.g., local events and places worth recording, local cultural habits such as shops opening hours), and the current status of the local world (e.g., local traffic conditions and whether the current day is a working day or a holiday). The LBS also needs to know existing correlations between context data and user/service data. This information identifies e.g. user preferences and service accessibility that are context-dependent.. These specifications form the context module, while factual context data is dynamically acquired from given data sources, e.g. weather data is extracted from the web pages of the local meteorological station. For each data source, we call context profiles the description of the locally available context data. They may contain, for example, a set of URLs with associated description of which kind of data is available and how it can be extracted, or a pointer to a tourist office database or set of XML files such as an event schedule.

**Context Profiles Matcher.** Similarly to the other matchers, this one is responsible for establishing and maintaining the mapping between the context module and the context profiles.

**Query Profiles and Query Relaxation Profiles.** The components in the basic infrastructure are those needed to understand user queries, refine it using contextual and user knowledge, and identify services that may match user's interest. Additional functionality may be achieved in terms of smarter query processing and calls for additional information that we define hereinafter as query profiles and query relaxation profiles (not shown in Figure 1). A query profile holds for each user query a description of its successive reformulations computed by the multi-step query processing strategy, as well as the relevant subsets of the user profile (conveying the user data that has been found relevant for this specific query) and of the context data (conveying the contextual data that has been found relevant for this specific query). Query profiles are stored within the LBS as element of a sequence of queries that we makes up a user interaction, i.e. an exchange between the user and the LBS that leads the user, through a series of questions & answers, to get the desired information. In addition, query relaxation is a very common topic in LBS. Whenever the user needs additional results or a perfect matching can not be accomplished, a query relaxation strategy is activated. The strategy calls for user specifications about what can be relaxed (i.e. which selection criteria can be softened to a larger selection) and which ordering of relaxations is to be preferred. We call query relaxation profile this user-driven and query-specific specification. The analysis of query profiles can assist to refine the relaxation rules and ranking functions for better recommendation and more efficient query relaxation.

## 7 Interactions between Components

This section briefly shows a usage scenario to illustrate the interplay of the different components in the data infrastructure. We first show a set-up scenario, followed by a query scenario.

**Set-up Scenario.** Let us assume a software company has an LBS skeleton, i.e. all the software to run the planned location-based services, and wants to set-up a first version of its LBS servicing tourists visiting the city of Lausanne. Setting up this specific LBS is a knowledge acquisition process. As we already mentioned, the first task is to find and import one of the existing ontologies for the tourism domain. The imported concepts form a first draft for the service module, generically describing standard services in support of traveling tourists. The second initialization step is acquisition of the local context. This can be achieved through import of data files acquired from local providers (e.g. the local tourism-related organizations such as cultural associations, local press, movie distributors, transport companies, and so on). Alternatively, data can be captured from public websites using knowledge extraction techniques (e.g. [8]). Captured and acquired data are formatted to define and populate the context module. Such a priori knowledge of context may be needed to perform the following step, which is acquisition of service descriptions from local service providers. As stated, we assume that local providers will make their service descriptions available, not necessarily following a fixed format or adhering to a fixed terminology. This is where the LBS will start building the shared terminology, in its attempt to understand what a given service description means. For example, retrieving the service description term from WordNet and associating it to the corresponding term already in the service module. Acquiring service knowledge is a sophisticated task, as the goal is not to import service descriptions but to build the abstract view of services that forms the service module. This knowledge abstraction step relies on linguistic techniques (to identify major relevant terms) as well as on semantic techniques (to only retain what is useful in differentiating the service from the other services). Building the service module obviously includes building the mapping between the module and the service profiles. Once the service and context modules are set up, the LBS is ready to start receiving queries from users. Here the question is whether user profiles will come from the user device or will have to be retrieved from some external repository of user profiles. Given the sensitivity of the issue, the former is likely to prevail. This means the LBS has to run a user profile understanding process, equivalent to the service profile understanding process. However, some generic knowledge about users characteristics can be initialized a priori within the user module. Actual user profiles will be matched against this initial set-up, the matching possibly leading to enriching the user module and the shared terminology.

**Query Scenario.** Here we just sketch what happens in the proposed LBS. User queries in our approach are formulated by stating which kind of service the user is interested in, and the spatial (e.g. proximity) and temporal (e.g. current availability) conditions that are to be taken into account while selecting specific services in response to the query. The user query can also specify thematic (non-spatial and non-temporal) conditions to refine the search for services of the given kind. The hypothetical query "Give me nearby restaurants still serving Swiss traditional cuisine after 2pm today" denotes restaurant as the desired a kind of service, together with the specification of spatial, temporal, and thematic conditions. Query processing within the LBS includes the following phases: 1) understanding the query (are the terms and the service they denote known in the core ontology or in the shared terminology? If not, search external ontologies); 2) retrieving from the service module the prototypical description of this kind of services in order to check preconditions for using services of this kind and check that conditions stated in the query can be actually evaluated (e.g. that the type of cuisine offered by a restaurant is known); 3) personalizing the query, i.e. check if relevant knowledge in the user profile allows refining the query, e.g. refining a query for a restaurant into a query for a restaurant within the price-range defined in user's preferences; 4) contextualizing the query, i.e. check if context data allows further refinement, e.g. replacing the qualitative expression "cheap restaurant" with the quantitative expression "restaurant with average price less than 40 CHF"; 5) executing the query, i.e. find relevant services in the order of preference, if any, as stated in the reformulated query. Notice that the order in which to execute phases 2, 3, and 4 can be changed without influencing the final result.

## 8 Conclusion and Future Work

In this paper, we first discussed the features characterizing data management in a LBS, i.e. locality, mobility, dynamics, heterogeneity, on the fly interoperability, and modularity. We emphasized the split in data organization between knowledge that is elaborated by the LBS itself (the core ontology and the shared terminology) and knowledge that is provided by external partners (users, context, and service providers). The core ontology represents the domain-specific conceptual views in a unified fashion. In addition, the shared terminology provides terminological support to overcome the heterogeneity problem that often results from the diversity of the cultures and languages, as well as from the lack of the knowledge of the LBS' structure and naming for users.

Our modular data infrastructure is mainly based on OWL (W3C recommended Web Ontology Language). Its full-fledged support to processing queries and reasoning over a large scale of ontology repositories still calls upon the convergence of many efforts from both the semantic web community and industrial practitioners. We can envision many potential challenges in deploying the future LBS, e.g. defining a robust ontology query language, developing more powerful ontology management tools, enhancing reasoning capabilities on complex data types such as spatio-temporal data etc.

## References


1. Bernstein, P.A.: Middleware: a model for distributed system services. Communications of the ACM 39(2), 86–98 (1996)
2. Cole, A., Duri, S., Munson, J., Murdock, J., Wood, D.: Adaptive Service Binding Middleware to Support Mobility. In: Proceedings of the 23rd International Conference on Distributed Computing Systems Workshops (ICDCSW 2003) (2003)
3. Jacobsen, H.-A.: Middleware for Location-Based Services. In: Schiller, J., Voisard, A. (eds.) Chapter of the book Location-Based Services, Morgan Kaufmann Publisher, San Francisco (2004)
4. Jensen, C., Friis-Christensen, A., Pedersen, T., Pfoser, D., Saltenis, S., Tryfona, N.: Location-Based Services - A Database Perspective. In: Proceedings of the Eighth Scandinavian Research Conference on Geographical Information Science, pp. 59–68 (2001)
5. Parent, C., Spaccapietra, S., Stuckenschmidt, H. (eds.): Ontology Modularization. Springer, Heidelberg (to appear, 2008)
6. Schmidt-Belz, B., Laamanen, H., Poslad, S., Zipf, A.: Location-based Mobile Tourist Services - First User Experiences. In: Proceedings of International Conference for Information and Communication Technologies in Travel and Tourism (ENTER), Helsinki, Finland (2003)
7. Shafer, S., Krumm, J., Brumitt, B., Meyers, B., Czerwinski, M., Robbins, D.: The New Easyliving Project at Microsoft Research. In: Proceedings of the Joint DARPA/NIST Smart Spaces Workshop (1998)
8. Tezuka, T., Lee, R., Kambayashi, Y., Takakura, H.: Web-Based Inference Rules for Processing Conceptual Geographical Relationships. In: Proceedings of the 2th International Conference on Web Information Systems Engineering WISE (2001)
9. Want, R., Hopper, A., Falcao, V., Gibbons, J.: The active badge location system. ACM Transaction on Information Systems 10(1), 91–102 (1992)
10. Yu, S., Spaccapietra, S., Cullot, N., Aufaure, M.-A.: User Profiles in Location-based Services: Make Humans More Nomadic and Personalized. In: Proceedings of IASTED International Conference on Databases and Applications, Innsbruck, Austria (2004)
11. Yu, S.: Contextualized and Personalized Location-based Services. Thesis No. 3896, EPFL, Switzerland (2008)

# A Method to Derivate SOAP Interfaces and WSDL Metadata from the OGC Web Processing Service Mandatory Interfaces

Gonzalo Sancho-Jiménez, Rubén Béjar, M.A. Latre, and Pedro R. Muro-Medrano

Computer Science and Systems Engineering Department, University of Zaragoza, María de Luna 1, E-50018 Zaragoza, Spain
{gsancho,rbejar,latre,prmuro}@unizar.es

**Abstract.** Web Processing Services (WPS) expose processing functionality using Web Service technology. The WPS specification describes the interfaces to publish geospatial processes on the Web. It includes a platform-neutral and several platform-specific versions of its interfaces. Some of the platform-specific interfaces are mandatory and others are optional. In this paper, we present a method to support the automatic derivation of the optional SOAP interfaces and WSDL metadata from the mandatory ones in any WPS. These interfaces can then be used to facilitate the chaining of the WPS with other Web Services, using for example BPEL, and to improve the interoperability of these services. In addition to that, we have created a tool to validate the proposed method.

## 1 Introduction

The Open Geospatial Consortium (OGC) Web Processing Services (WPS) specification [1] describes the interfaces needed to design a service in which geospatial process is offered on the Web. It includes a platform-neutral and several platform-specific versions of its interfaces. Some of the platform-specific interfaces are described as mandatory while some others are optional, nevertheless offering support to all of them is recommended. At the same time, multiple encoding ways for requests and responses are described, each of them appropriate for one or more platforms.

The aim of this paper is to propose a method to support the automatic derivation of the optional Simple Object Access Protocol (SOAP) interfaces and Web Services Description Language (WSDL) metadata from the mandatory ones in any WPS, as well as a tool to test its viability. These interfaces can then be used to facilitate the chaining of the WPS with other Web Services, using for example Business Process Execution Language (BPEL).

This method helps developers to avoid the tedious task of programming the different platform-specific interfaces to the geoprocessing operations encapsulated by the WPS. They only need to create the mandatory ones and the others

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 375–384, 2008.

are automatically generated. It also helps to avoid mistakes and unwanted differences among the different platform-specific interfaces to the same service. Then, it will be easily maintained and adapted to specific requirements.

The method can be applied to any compliant OGC WPS because they must support at least the mandatory parts of the specification. It does not require access to the source code or the binaries of the WPS, it only needs network access to the mandatory interfaces of the WPS.

The paper shows how the mandatory platform-specific interfaces define a model of the service, that can be used to automatically create actual instances of the service that support different technologies. The mandatory interfaces could be seen as the "canonical" specification which is used to generate its many optional interfaces, facilitating thus interoperability of the WPS. This would prevent problems coming from different behaviours of the service under its different interfaces.

The rest of the paper is organized as follows: Section 2 summarizes related work including a WPS standard description. Section 3 describes the automatic derivation method to offer SOAP/WSDL support in a WPS. Then, Section 4 tests the feasibility of the method showing an implemented tool and giving an example of use of this tool. The last sections introduce some ideas on future work and conclude.

## 2 Related Work

There is a wide set of published works dealing with the issue of SOAP/WSDL adaptation method for OGC services. Those works use automatic adaptation methods which must then be checked.

In [2] there is an analysis of the use of BPEL in comparison to Web Services Modelling Ontology (WSMO) and a state-of-the-art composition approach. It states that current implementations of OGC services are some kind of hybrid Representational State Transfer (REST)-based services, whereas BPEL requires SOAP services, so OGC services, which do not provide SOAP interfaces, need a wrapper, which acts as a proxy to the OGC services. In this paper we review this wrapping method in order to support the automatic derivation from the optional interfaces to the mandatory ones.

The suitability of the Web Service Orchestration (WSO) technology as a possible solution for disaster management scenarios has been evaluated in [3]. It analyses and describes data format adaptation for OGC services with a proxy server acting as binary transcoding service. The proxy launches an OGC compliant request to the OGC service and receives the response from the service. The proxy stores the physical part of the response on a simple http-server directory and returns the Uniform Resource Identifier (URI) of the stored data to the BPEL process. Using a proxy method is a suitable idea for automatic adaptation. E The first steps in developing a Web service framework for heterogeneous environmental information systems are presented in [4]. With a framework which allows an easy assembly of data and method services which are distributed over the entire Web. The framework [4] uses the Web Service Framework, which allows users to create their own services by combining the existing ones. This framework uses a generic adapter which allows a technical encapsulation of different middleware technologies.

The aim of [5] is to present a discussion of technology issues considered in an initiative of the OGC Interoperability Program. It is focused on creating general recommendations and guidelines for WSDL/SOAP support to existing OGC Web Services. In this paper these recommendations are followed as much as possible.

### 2.1 The WPS Standard

The specified WPS provides client access to pre-programmed calculations or computation models that operate on spatially referenced data. The data required by the service can be delivered across a network, or be available at the server. The info can be provideded as image data or through data exchange standards such as Geography Markup Language (GML). The calculation can be as simple as subtracting one set of spatially referenced numbers from another (e.g. determining the difference in influenza cases between two different seasons), or as complicated as a global climate change model [1].

The WPS interface specifies three operations that can be requested by a client and performed by a WPS server and that are considered as mandatory and must be implemented by all WPS servers. These operations are: `GetCapabilities`, which allows a client to receive service information, `DescribeProcess`, which should return detailed information about server's processes and `Execute`, which runs a specified process implemented by the WPS. The mandatory platform-specific interface of these operations must be implemented and the optional platform-specific interface is recommended to be supported.

In table 1, each operation includes its optional and mandatory platform-specific interfaces. A compliant WPS must contain the mandatory platform-specific interface, although the implementation of the optional ones would improve interoperability with different clients.

**Table 1.** Optional and mandatory platform-specific interfaces

| Operation | GET/KVP | POST/XML | SOAP |
|---|---|---|---|
| `GetCapabilities` | Mandatory | Optional | Optional |
| `DescribeProcess` | Mandatory | Optional | Optional |
| `Execute` | Optional | Mandatory | Optional |

The WPS standard recommends including WSDL metadata for each operation. Despite being less comprehensive mechanism, WSDL makes dynamic binding to dynamic services easier. It offers a widely adopted alternative to the publishing mechanism built into the WPS interface specification.

The aim of this paper is to describe a method in which only the mandatory platform-specific interface has to be defined, while the others may be automatically derived, allowing to add optional platform-specific interfaces to a WPS without requiring access to its source code or binaries.

This method allows all WPS to comply with World Wide Web Consortium (W3C) standards using SOAP/WSDL without code modification [5]. The reason for this mapping is the fact that certain developments only focus on supporting SOAP services and do not define GET and POST bindings. The reason for this is that this work heavily relies on mainstream standards such as Security Assertion Markup Language (SAML), eXtensible Access Control Markup Language (XACML) and the WS-* family which is oriented to support SOAP.

## 3 Automatic Derivation Method

In order to offer SOAP/WSDL support, we propose a method which combines several standardization efforts with a flexible and extensible Web service framework, supporting a higher interoperability between different resources.

It is necessary to create an intermediate proxy which adds a new SOAP/WSDL interface to the WPS that does not offer it. This proxy could remain separated or embedded in the processing service, being able to connect with the WPS through the mandatory interfaces. The proxy design consists of two modules. The first one generates the WSDL metadata through WPS request obtained data, and the second one adapts a SOAP request (generated from a WSDL metadata) to the mandatory WPS request.

Carrying out the adaptation task through a proxy allows us to incorporate it to the implementation of a WPS, no matter if it is under our control or not. If it is decided to use the adaptation proxy as part of a WPS which belongs to us, we can avoid the tedious chore of having to program the different interfaces over each one of the offered operations: it is only necessary to program the mandatory operations, which are the ones that provide more information.

Figure 1 shows an example of a proxy integration into a WPS in order to define a new SOAP interface. The mentioned WPS contains only a definition of the offered services and only an interface per public-accessible operation. The proxy

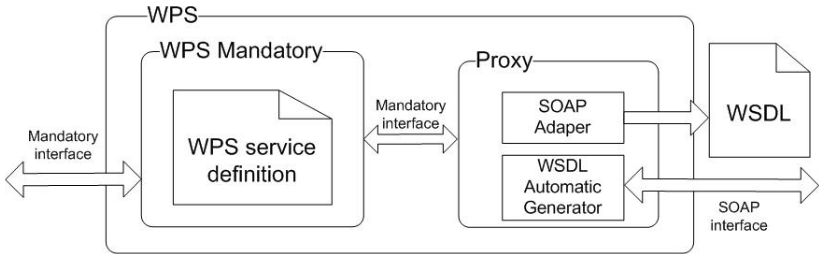



**Fig. 1.** Example of proxy integration

accesses internally (through the mandatory interface) to the WPS operations offering a new WSDL/SOAP interface.

### 3.1 WSDL Automatic Generation

The WSDL automatic generation method provides the WSDL metadata document by using data obtained WPS requests, following through the WPS structure as far as possible. The WSDL generated describe the WPS SOAP interface to provide it over the Internet. A client program connecting to a WPS can parse the WSDL to determine what functions are available on the server.

The WSDL generation request for a WPS is sent directly to the proxy URL which does not know the WPS URL. That is the reason why the WPS URL needs to be included as a parameter of the request. With this URL the proxy may perform the necessary requests against the WPS using the mandatory interface to compose the WSDL.

Following, as far as possible, the best practices of the OGC WPS 1.0.0 specification, the WSDL service description can be requested as a single WSDL document (including all the operations) or as a document per operation. An example of a WSDL request for all the WPS operations as a single WSDL document: `http://URLProxy?URLWPS=http://URLwps&WSDL`.

Attending to the WSDL request of an individual process, it keeps the same structure including the operation identifier as in `http://URLProxy/identifier?URLWPS=http://URLwps&WSDL`.

Each WSDL request contains a `GetCapabilities` and a `DescribeProcess` methods specification, but as many `Execute` method specifications (with their own parameters) as operations offered in the WPS. Figure 2, shows that `GetCapabilities` and `DescribeProcess` requests contain the parameters described in the WPS specification but are codified as SOAP parameters. The `GetCapabilities` operation includes (in WSDL document) the parameters `AcceptVersions` and language and in case of the `DescribeProcess` operation, version, language and identifier are included.

The responses received from these two operations comply with the XML Schema Definition (XSD) defined in the standard but with the difference that these responses are wrapped as a SOAP response.

In order to fit the WPS request optimally and in an automatic way, each of the parameters of the `Execute` operation have to be defined as much as possible in the WSDL document. The `DescribeProcess` response is parsed to obtain the necessary information required to generating the WSDL `Execute` request and response; each input and output is codified as a different element in the SOAP body using the identifier as name and the most similar type defined in the WSDL specification. The operations containing `state=true` are enabled, but they are added as if they do not have that option; this is due to the fact that, in the prototype, no mechanism allowing this request is implemented while the option of allowing the calculation is intended.

Each of the `DescribeProcess` response parameters are included into the response and are reconverted into a WSDL element, in which the number of

```
<xsd:element name="GetCapabilities">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="acceptedVersion"
     type="xsd:string"/>
      <xsd:element minOccurs="0" name="language" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:elememnt name="DescribeProcess">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="1" name="identifier"
     maxOccurs="unbounded" type="xsd:string"/>
      <xsd:element minOccurs="1" name="version" type="xsd:string"/>
      <xsd:element minOccurs="0" name="language" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

**Fig. 2.** `GetCapabilities` and `Describe` method specification in the WSDL generated

occurrences is specified with a maximum and a minimum limitation. This element contains a `complexType` field composed with each one of the `DescribeProcess` response attributes specifying their possible and default values. Code example shows an automatic conversion from WPS `ComplexType` specification to WSDL specification.

As seen in Figure 3, additional information is included specifying, for example, diverse parameters in the request attributes. In this example, just with the WSDL specification, it is possible to know that in the `complexData` attribute, it has to be specified an encoding attribute as well as a schema attribute (which has a default value); it can also be deduced that the output data are to be codified in a 64 base, and also, that the attribute may appear zero or more times in the request.

```
<complexType name="IDOperation_IDParameter_complex">
  <sequence>
    <element name="encoding" type="xsd:anyURI"/>
    <element name="schema" type="xsd:anyURI"
   default="http://schemas.opengis.net/wfs/1.1.0/wfs.xsd" />
    <element name="data" type="xsd:base64Binary"/>
  </sequence>
</complexType>
<element name="IDOperation_IDParameter_type">
  <complexType>
    <sequence>
      <element name="input"
     type="tns:IDOperation_IDParameter_complex" minOccurs="0"
     maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>
```

**Fig. 3.** `Execute` method specification in WSDL

### 3.2 Automatic SOAP Adaptation

Once the WSDL has been generated, this can be used by any user as a description to generate requests to the SOAP WPS interface. The requests are sent to the proxy URL transparently to the user. The WPS URL is contained in the proxy URL like a parameter towards which the operation is finally going to be performed as seen below:

```
<wsdl:service name="OperacionesService">
  <wsdl:port name="\textbf{ServicioOperaciones}" binding="impl:WPS_Binding">
    <wsdlsoap:address
   location="http://URLProxy?URLWPS=http://URLwps"/>
  </wsdl:port>
</wsdl:service>
```

The proxy parses the SOAP request to obtain its attributes, values and the WPS URL. Once it is done, the proxy generates the request following the mandatory interface and sends it to the WPS. For example by POST in case of `Execute`, and by Key Value Pairs (KVP)/GET in case of `GetCapabilities` and `DescribeProcess`. When the WPS has performed the required operations, the response is generated and sent to the proxy. This proxy parses the XML parameters and generates a SOAP response that corresponds to the one defined in the WSDL document.

The automatic adaptation SOAP service can be located in a different server to the WSDL generator due to fact that they are two independent services. SOAP adapter works independently and without having information about the request WSDL description which is going to adapt, making a streaming adaptation. This automatic adaptation is possible thanks to the above guidelines on the generation of WSDL description types, since these types contain enough information to generate a valid request to the WPS from the SOAP request.

## 4 Testing Tool

Following the described method, a tool has been designed to test its viability. This tool provides the mentioned features and can be used to add new WPS platform-specific interfaces to any WPS.

As shown in Figure 4, the proxy generates the WSDL metadata in accordance with the WPS and performing the request conversion between SOAP and the mandatory WPS platform-specific operation. The proxy is completely transparent for the WPS client, so it can use it as another service interface.

The first steep is the WSDL generation: the proxy needs to know all the operations that the WPS offers in order to generate the WSDL description; with this aim a `GetCapabilities` request is made using the standard (GET/KVP). Once all the operations are known, a `DescribeProcess` request is performed (GET/KVP) per each operation to know all operation parameters. With these data and a collection of guidelines justified above, it is possible to build a WSDL in an automatic way; the second steep is SOAP adaptation: the tool simply follows the proposed method to SOAP encoding and decodes the requests and responses, providing that WSDL metadata has been well defined.

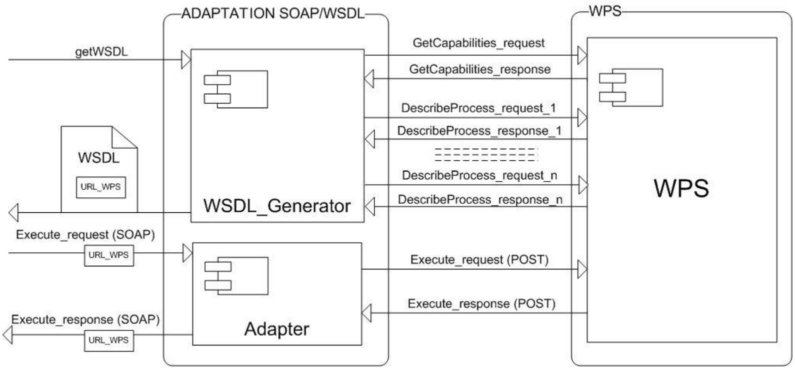



**Fig. 4.** Proxy design

### 4.1 Composition Example

To give an example of the use of this tool, a WPS without SOAP interface has been adapted in order to be chained with BPEL engine. The final target operation is an intersection between a vectorial and a raster from several operations of the same WPS. This series of operations can be chained in a simple and fast way, allowing creating more complex operations with scalability. For the BPEL composition, only `Execute` methods of each sub-operation are needed, since `GetCapabilities` and `DescribeProcess` methods list and describe the operations.

The Spanish Spatial Data Infrastructure (IDEE) WPS are used with the purpose of testing this tool. It has been implemented using the degree [6] framework which uses the 0.4.0 WPS version. This implementation does not provide the optional SOAP/WSDL interfaces. IDEE develops the operations `vectRastConversion`, which converts a vector data to a raster data, and `rasterIntersect`, which intersects two raster maps in a new one.

In order to make the BPEL's composition operation has been chosen the use of Apache Orchestration Director Engine (ODE) [7] which executes business processes written following the BPEL standard. It talks to Web services, sending and receiving messages, handling data manipulation and error recovery as described by the process definition. It supports both long and short living process executions to orchestrate all the services that are part of an application.

In Figure 5 a scheme of the operation desired to make can be seen. In the first WPS operation the conversion from a vectorial map to a raster one is made, next the intersection with other raster is done. These operations can be independently reused as belonging to a complex processing service.

BPEL's process that composes the operations in Figure 6 consists of a sequence that includes all the described operations since it is a not parallelizable operation. The results of each sub-operation are assigned to the inputs in the next operation with the necessary parameters that are defined in the process input. Each one of the sub-operations makes a SOAP request to an adapter in an

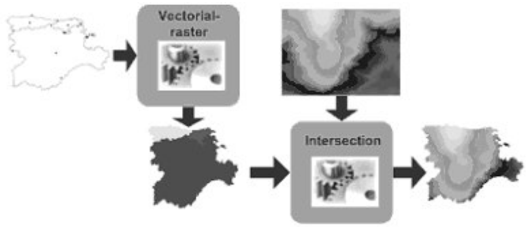



**Fig. 5.** Schema of the operation

automatic way and following the WSDL scheme that has been used to make the composition. In the adapter server, each SOAP request is processed in different instances. In each instance, the requests are decoded and the POST `Execute` request is composed to be sent to the specified WPS. When the WPS execution has been finished a POST response is sent to the Adapter which encodes the SOAP response and sends it to the BPEL process to continue the execution.

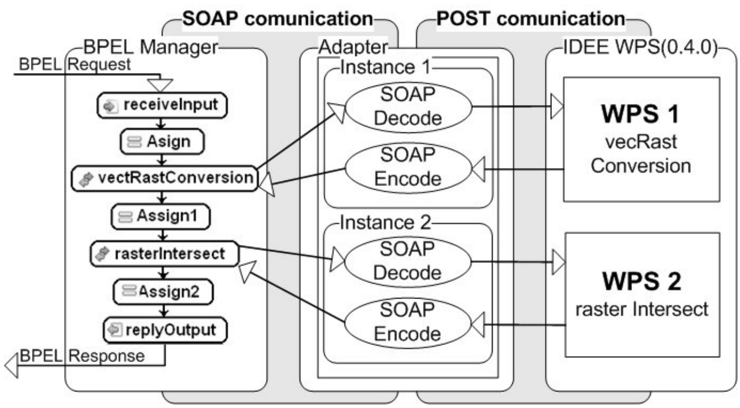



**Fig. 6.** BPEL's process

## 5 Conclusions

Processing services provide a new way of offering geoprocessing capabilities adapted to different user needs. Interaction among processing services is possible, in spite of the fact that they are independent, providing a new way to create new complex geoprocessing operations.

The WPS standard opens the possibility to implement several platform-specific interfaces as SOAP/WSDL needed by mainstream standards such as SAML, XACML and the WS-* family. The method presented in this paper allows adding the optional SOAP platform-specific interfaces to any WPS, without requiring access to the source code or the binaries of this service; it only needs network access to its mandatory interfaces. Using this method in order to provide optionals interfaces, it is possible to avoid errors that could be committed when programming different platform-specific interfaces to offer the same functionality. It also makes it easier to integrate WPS with other Web Services, facilitating thus interoperability.

In spite of the fact that the described method follows as much as possible the OGC SOAP recommendations [5], new additional information has been added using all WSDL and SOAP implementation resources in order to approach SOAP and mandatory requests. For example, input parameter types, units of measure, default values, formats, schemes and encodings have been added instead of an undefined list of String parameters as recommended [5]. It allows automatic derivation but it also allow increasing the WSDL description information provided. To allow SOAP adaptation without information stored, the WPS URL has been added to the SOAP description, consequently to the SOAP request that has been added too.

A tool has been designed to validate the presented method. The designed tool works correctly and, to test it, a BPEL composition, using IDEE WPS, has been implemented due to the fact that IDEE WPS has not got any SOAP/WSDL interface. The incorporation of this tool in IDEE WPS is currently being studied.

## Acknowledgements

The basic technology of this work has been partially supported by the Spanish Ministry of Science and Technology through the project TIN2007-65341 from the National Plan for Scientic Research, Development and Technology Innovation, by the University of Zaragoza through the project UZ200700024 and by GeoSpatiumLab S.L.

## References

1. Open Geospatial Consortium: Web Processing Service (WPS), version 1.0.0 (2007), http://www.opengeospatial.org/standards/wps
2. Gone, M., Schade, S.: Towards Semantic Composition of Geospatial Web Services – Using WSMO in Comparison to BPEL. In: Proceedings of GI-Days 2007, Münster, Germany, pp. 43–63 (2007)
3. Weiser, A., Zipf, A.: Web Service Orchestration (WSO) of OGC Web Services (OWS) for Disaster Management. In: Proceedings of the 3rd International Symposium on Geoinformation for Disaster Management, Toronto, Canada, pp. 33–41 (2007)
4. Radetzki, U., Alda, S., Bode, T., Cremers, A.: First Steps in the Development of a Web Service Framework for Heterogeneous Environmental Information Systems. In: Proceedings of the 16th International Conference on Informatics for Environmental Protection, Vienna, Austria, pp. 43–63 (2002)
5. Open Geospatial Consortium: OWS 5 SOAP/WSDL Common Engineering Report version 0.1.0 (2008), http://www.opengeospatial.org/standards/dp
6. OSGeo Foundation: Deegree (2008), http://www.deegree.org/
7. Apache Software Foundation: Orchestration Director Engine (ODE) (2008), http://ode.apache.org/

# Managing Sensor Data on Urban Traffic

Claudia Bauzer Medeiros[1], Marc Joliveau[2], Geneviève Jomier[3], and Florian De Vuyst[2]

[1] IC, University of Campinas, UNICAMP, Brazil
[2] Ecole Centrale de Paris, France
[3] University Paris Dauphine, France
cmbm@ic.unicamp.br, {marc.joliveau,florian.de-vuyst}@ecp.fr, genevieve.jomier@dauphine.fr

**Abstract.** Sensor data on traffic events have prompted a wide range of research issues, related with the so-called ITS (Intelligent Transportation Systems). Data are delivered for both static (fixed) and mobile (embarked) sensors, generating large and complex spatio-temporal series. Research efforts in handling these data range from pattern matching and data mining techniques (for forecasting and trend analysis) to work on database queries (e.g., to construct scenarios). Work on embarked sensors also considers issues on trajectories and moving objects.

This paper presents a new kind of framework to manage static sensor data. Our work is based on combining research on analytical methods to process sensor data, and database procedures to query these data. The first component is geared towards supporting pattern matching, whereas the second deals with spatio-temporal database issues. This allows distinct granularities and modalities of analysis of sensor data in space and time. This work was conducted within a project that uses real data, with test conducted on 1000 sensors, during 3 years, in a large French city.

## 1 Introduction

This paper analyzes problems in managing spatio-temporal data obtained at real time, from a large network of urban traffic sensors. These sensors continuously capture distinct kinds of data on traffic, to be used for analysis in traffic management and planning. At all times, experts must also take into account sensor failures, to avoid incorrect decisions.

Intelligent Transportation Systems (ITS) research is multidisciplinary, encompassing people from many distinct areas – from computer scientists to psychologists. The goal is to improve the overall transportation infrastructure offered to all kinds of travelers – from those who need to cross a street to overseas flight passengers. Improvements can be felt at distinct levels – increased safety, economic gains or efficiency of the transportation system itself. The design and development of software embarked in intelligent vehicles, sensitive to GPS positioning, is another area in which geospatial information management is adopted.

Computer Science research in ITS spatio-temporal patterns is largely dedicated to either algorithmic manipulations of objects on road networks, or on

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 385–394, 2008.

data structures and models to process and store information on mobile objects. A large percentage of this research, moreover, concerns mobile sensors embarked on vehicles, providing information on vehicle movement. Both research trends require sophisticated models and algorithms to process spatio-temporal queries, and to support ITS requirements.

As will be seen, our work is based on a different kind of approach. Rather than manipulating the original sensor spatio-temporal data series, we first preprocess these data, eliminating noise, filling missing values, and reducing their dimensionality. The result are clean series with nice properties. Next, these data are stored in a DBMS, on which ITS queries and pattern analyses can be performed. Our work thus combines research results in spatio-temporal databases to those in mathematical modeling of complex temporal series. We do moreover take into account the influence of human activity in urban areas (e.g., street markets, or accidents) to derive better evaluation of traffic conditions.

Section 2 presents an overview of the context of our work. Section 3 presents our solution for preprocessing and summarizing sensor produced data. Section 4 discusses the spatio-temporal queries that can be posed in our framework. Section 5 briefly comments on related work, and section 6 concludes the paper.

## 2 Overview of the Problem

Our research is conducted within the CADDY project (Control of the acquisition and storage of massive temporal Data volumes and DYnamic models) [1]. CADDY involved a multidisciplinary research team composed of computer scientists and experts in traffic management and planning. The goal was to develop a computational framework for decision support in urban traffic management.

The sensors used in our research are fixed along street networks, and continuously collect and broadcast several kinds of data on traffic movement. Data are sent to stations, and forwarded to a central data storage facility. Two major traffic variables are used by experts in this context, producing two distinct (but interdependent) spatio-temporal series – see section 3 for their relationship:

- the vehicle flow rate ($q$), i.e., the number of vehicles that have passed in front of the sensor for a given time period, usually minute or hour;
- the occupancy rate ($\tau$), i.e., the average space between vehicles in a given time period. An occupancy rate of 100% means there is no space between vehicles, while 0% means no vehicles.

Figure 1 shows the flow rate and average occupancy in a weekday. One can see traffic peaks between 8 and 10 AM, for instance. Our source data cover 1000 sensors for 3 years, where each sensor collects data every three minutes. Each day is delivered in a separate set of files (for $q$ and $\tau$), for a grand total of approximately 420x$10^6$ values. Our data were provided by the CLAIRE traffic supervision system [2] from INRETS (the French National Transportation Research Institute). CLAIRE models an urban street network through an oriented graph, where each edge corresponds to a street segment. Sensor data are provided on each edge, associated with the corresponding spatial location.

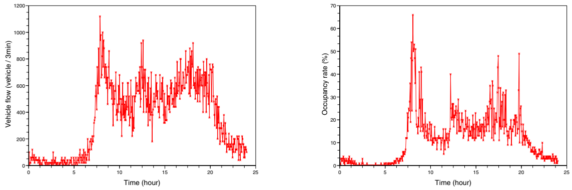



**Fig. 1.** Flow and occupancy rates, for one sensor, in a given street segment

Processing these data presents several challenges, since $q$ and $\tau$ must be combined to allow experts an overall view, complicating the mathematical analysis. Data are kept separate for each sensor and each day. This means that there are at least two kinds of long series that can be constructed – (a) per sensor, over time, and (b) per timestamp, for all sensors. Each kind – (a) or (b) – supports a distinct analysis. The first allows studying the behavior, for one sensor, through time (fixed point in space, varying time), while the second presents a snapshot of the entire network, at a given timestamp (fixed time, varying space). Joint analyses need to correlate these factors, resulting in a complex multidimensional data space, as will be seen in section 3.

Another issue is quality – the data transmitted by the sensors are very noisy, and contain many gaps, mainly due to sensor failures and/or network breakdowns. Thus, the data must be cleaned before they can be processed. The next section presents our solution for the dimensionality and missing value problems. It shows how these two series can be cleaned, and represented with very small errors by a compact descriptor (thus reducing dimensionality).

## 3 Summarizing Sensor-Based Data

This section presents our solution to pre-process the spatio-temporal sensor data series, summarizing them and reducing their dimensionality. It starts by presenting a method that dramatically reduces the dimensionality of each series ($q$ and $r$), for all sensors, while managing to maintain almost all the original information. Next, it shows how to derive missing values. Finally, it combines both kinds of series (flow rate and occupancy), to describe the overall state of each sensor per day, mapped to the interval [0,1]. This interval can be divided into several classes, which represent specific traffic states defined by experts (e.g., "heavy", "bottleneck"), thus supporting subsequent queries not only on values but also on semantically meaningful states.

### 3.1 Reducing Dimensionality – The STPCA Method

In [3], we introduced the Space-Time Component Analysis (STPCA), a new method to develop descriptors of spatio-temporal data series. This method is based on applying Principal Component Analysis (PCA) [4] to both spatial and temporal dimensions, as follows.

Assume that there are data available for $N$ days. Data collected on the flow rate or occupancy rate are stored in a matrix $\mathbf{X}^d$, where $d$ symbolizes the day. The time series corresponding to measurements collected by sensor $i$ at day $d$ is given by row $\mathbf{x}_i^d$. The following steps are applied separately to flow rate and to occupancy rate, obtaining two sets of time series processed by STPCA.

1. Assemble day matrices horizontally, for spatial analysis, in a single matrix $\mathbf{Y}$, and vertically for temporal analysis in a matrix $\mathbf{Z}$. In matrix $\mathbf{Y}$, each column contains all values obtained in a timestamp, and each row corresponds to one sensor, with values varying through time, through all days (fixed location). In matrix $\mathbf{Z}$, columns are the timestamps of one day and each row corresponds to data from one sensor, for one single day (fixed time); there are as many rows for each sensor as there are capture days.
2. Compute singular value decomposition for matrices $\mathbf{Y}$ and $\mathbf{Z}$, as follows For spatial correlation matrix $\mathbf{M}^s = \mathbf{Y}\mathbf{Y}^T$, compute the $K$ first spatial eigenvectors $(\boldsymbol{\Psi}^k)_{k=1 \ldots K_M}$, with $K \ll K_M$ , storing them in matrix $\mathbf{P}$. For temporal correlation matrix $\mathbf{M}^t = \mathbf{Z}^T\mathbf{Z}$, compute the $L$ first temporal eigenvectors, with $L \ll L_M$, $(\boldsymbol{\Phi}^l)_{l=1 \ldots L_M}$, storing them in matrix $\mathbf{Q}$.

$$\mathbf{P} = col\left(\boldsymbol{\Psi}^1, \boldsymbol{\Psi}^2, \ldots, \boldsymbol{\Psi}^K\right).\mathbf{Q} = col\left(\boldsymbol{\Phi}^1, \boldsymbol{\Phi}^2, \ldots, \boldsymbol{\Phi}^L\right).$$

3. Finally, the STPCA estimate $\hat{\mathbf{X}}^d$ of a day matrix $\mathbf{X}^d$ is defined by:

$$\hat{\mathbf{X}}^d = \mathbf{P}\mathbf{P}^T\,\mathbf{X}^d\,\mathbf{Q}\mathbf{Q}^T.$$

We point out that the reduced order matrix is given by:

$$\mathbf{X}_d^r = \mathbf{P}^T\,\mathbf{X}^d\,\mathbf{Q},$$

of size $K \times L$ where $K$ and $L$ are chosen to be small. Experiments done with very small values of these parameters, namely, $K = L = 3$, corresponding to a reduction factor of order $10^4$, demonstrate the ability of STPCA to compute a good approximation. – see [3] for details.

Intuitively, STPCA summarizes the “average” typical traffic behavior for a sensor, over all days in a period. Figure 2 illustrates flow rate time series measured by 4 sensors randomly chosen (grey curves), and their STPCA estimate (black curves), for $K = L = 3$.

**Filling missing values** STPCA cannot be directly applied to a data set containing missing values. Our solution to this problem [5] is the following. We use the Expectation Maximization (EM) [6] algorithm to estimate separately the spatial correlation matrix $\mathbf{M}^s$ and the temporal correlation matrix $\mathbf{M}^t$. We compute a complete estimation of the data set by using the $k$ nearest time series of each time series. We then project this estimation on the principal component of $\mathbf{M}^s$ and $\mathbf{M}^t$ approximations. Our experiments [5] show that results obtained by STPCA on data sets with an incompleteness degree higher than 20% stay very close to those obtained by STPCA on the corresponding complete data sets.

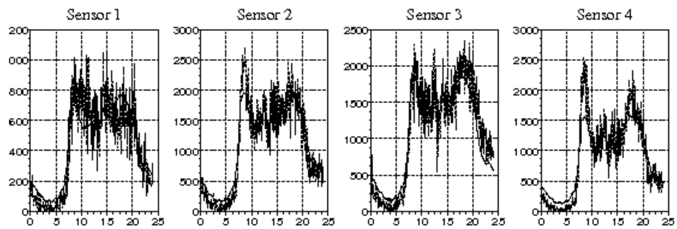



**Fig. 2.** Illustration of the flow rate time series, for one day, of 4 randomly chosen sensors. Original and STPCA time series using $K = 3$ and $L = 3$.

### 3.2 Attaching Semantics to Traffic Variables

Experts need to combine $r$ and $\tau$. To do that, we used the fundamental car-traffic law described in transportation theory, which defines the relation between $r$ and $\tau$. This relation built by approximating the set of couples $(r, \tau)$ observed at a sensor can be graphically represented by the fundamental diagram – the black curve on figure 3, which shows the correlation between $\tau$ (vertical) and $q$ (horizontal) rates, for one sensor, over one day.

We propose a new variable, the circulation state ($E$), derived from the fundamental diagram. This variable is based on computing two kinds of value for each sensor $i$, per day:

- the average maximum flow rate ($\tilde{q}_i$) is given by the mean of the maximum flow rate measured at sensor $i$ for each day. It corresponds to a near optimal flow rate value with respect to traffic in front of sensor $i$ ;
- the variable $\tilde{\tau}_i$ computes the daily average occupancy rate value when traffic reaches its maximum flow rate in a day.

At a fixed time $t$ for day $d$, the vehicle flow rate and the occupancy rate of a sensor $i$ are respectively given by $q^d_{i,t}$ and $\tau^d_{i,t}$. For each measurement, we compute $\theta^d_{i,t}$ the angle between the line given by $y = \tilde{\tau}_i$ and the line linking the points $(0, \tilde{\tau}_i)$ and $(q^d_{i,t}, \tau^d_{i,t})$. From $\theta^d_{i,t}$, we define $e^d_i(t)$ the value of the circulation state $E$ at sensor $i$ at day $d$ and time $t$:

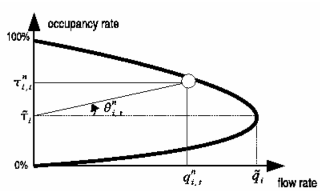



**Fig. 3.** Fundamental diagram with circulation state variable $E$

$$e_i^d(t) = 1/2 + 1/\pi \times arctg\Big(\frac{(\tau_{i,t}^d - \tilde{\tau}_i)}{100} \times \frac{\tilde{q}_i}{q_{i,t}^d}\Big).$$

We point out that values of $E$ are normalized, between 0 and 1. If the value of $E$ is close to 0, traffic is very calm and fluid (low flow and occupancy rates). A circulation state with a value of 1 represents an immobile congested circulation whereas a value close to 0.5 symbolizes a nearly optimal traffic with high flow rate. The circulation state thus combines two different variables into one without loss of information. Moreover, it gives a normalized and intelligible view of traffic.

Once we obtained these normalized values (per sensor, per day), we considered information from domain experts to define seven distinct traffic states based on the values of $e_i^d(t)$ – *Heavy*, *Fluid*, *Bottleneck*, etc., thereby transforming the numeric description into a symbolic description – e.g., see [7].

## 4 Querying and Mining Traffic Data

In [8] we described the structure of a data warehouse to store the raw sensor data. The design of this warehouse took into consideration the multiple spatial and temporal aspects that must be considered when dealing with traffic data, hence allowing aggregation along several dimensions.

We now extend this proposal by supporting the storage of all kinds of data discussed previously together with the raw data – i.e., the series with missing values filled, circulation ($E$) as well as STPCA summarizations (for $q$, $\tau$ and $E$), as well as their symbolic discretization. This multitude of data representations allows several kinds of analysis that are not reported elsewhere. This section indicates the classes of queries that can be posed. Though not implemented, they exemplify new kinds of analyses that we can obtain.

### 4.1 Pattern Analysis

Pattern analysis is deeply related with similarity search. The goal of similarity search is to find, in a database, all series which are similar to another series provided as input. Similarity mechanisms are based on the notions of *feature descriptor* and *distance function*. A feature descriptor is typically a set of values, summarizing the evolution of values in a series – e.g., in our case, STPCA. Two series are considered similar if the distance between their descriptors is below a threshold. Threshold values and distance functions must be validated by experts for each application domain. Storing the raw and clean series, and their STPCA representation, supports several kinds of similarity search, assuming that some distance function has been defined by the experts.

Let $I$ be an input series, which can represent $\tau$, $q$ or $E$ series. $Q1, Q2$ and $Q3$ are standard pattern analyses, whose implementation can be solved by similarity processing algorithms typical of data mining processes – see section 5. Thus, they can be processed on the cleaned data set using these standard procedures. $Q4$ through $Q7$, however, are new kinds of query. They can only be answered due to

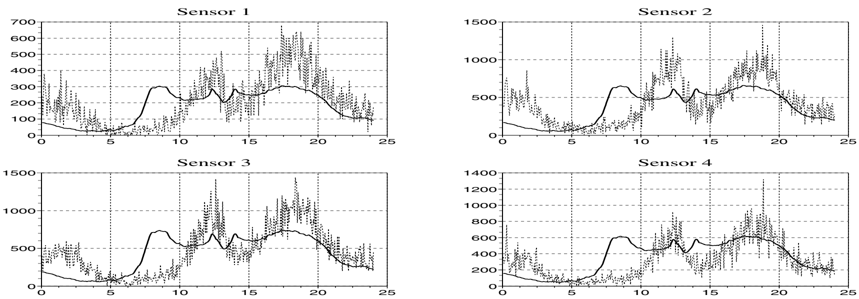



**Fig. 4.** Comparison of actual measured values for a sensor series (gray) and their STPCA estimate (black), for an atypical date - outliers

our spatio-temporal representations, e.g., a query on "typical behavior" requires STPCA.

– Q1 - Retrieve all series which are similar to $I$
– Q2 - Retrieve all series which are similar to $I$, but only for sensor $i$
– Q3 - Retrieve all series which are similar to $I$, but only for day $d$
– Q4 - What is the circulation behavior of sensor $i$ for day $d$ (on $E$)
– Q5 - What is the typical $q$ behavior of sensor $i$ (returns its STPCA estimate)
– Q6 - Show the sensors whose typical circulation behavior is closest to that of sensor $i$ on day $d$ (on STPCA applied to $E$)
– Q7 Show all sensors with a high margin of outliers for day $d$

Outliers can be detected by comparing a curve created by applying STPCA to $q$ or $\tau$, and the original series, for a given day and sensor. Figure 4 shows examples of such a difference – values computed for a typical day, for a sensor, and values captured on Christmas. The possibility of describing typical behavior allows checking for anomalies, and also detecting for which kinds of day an approximation must be tuned – e.g., eliminating these days from global matrices Y and Z, and computing STPCA for them alone.

### 4.2 Spatio-temporal Database Queries

Two kinds of spatio-temporal database queries can be considered – those on the different numeric series, and those on symbolic representations applied to $E$. The first can be divided into two basic situations – temporal and spatial queries; spatio-temporal queries are classically a result of the combination of both. Examples on queries on numeric series are

– Q8 - Spatial information – retrieve all series on $q$ (or $\tau$, or $E$) for sensors within a given area (i.e., a standard spatial window query that returns data on all sensors whose coordinates fall within the input region). This query returns a set of series (assuming each day is stored separately)

- Q9 - Temporal information – retrieve all series on $q$ (or $\tau$, or $E$) within a given period (i.e., a standard temporal query where the predicate will be checked against sensor timestamps). Screen copy on figure 5 is an example of the kind of output display of this query.
- Q10 – Spatio-temporal information – a combination of Q9 and Q10.

We finally come to queries on symbolic data. This allows users to retrieve information with associated semantics, closer to the user mental framework.

- Q11 – What are the sensors that on day $d$ had more than 30% of $Heavy$ circulation values

- Q12 – What are the sensors within a given region that in at least one day had more than 30% of $Heavy$ circulation values (spatial restriction to a region, and then executing $Q11$ repeatedly for each sensor in that region)

- Q13 – Which sensors within a given spatio-temporal frame had more than 30% of $Heavy$ circulation (combination of Q11 and Q12)

Last but not least, database queries can be combined with pattern matching queries using the symbolic representation, e.g., "$Q14$ – Which sensors within a given spatio-temporal frame had a symbolic circulation pattern similar to a symbolic input pattern".

## 5 Comparison to Related Work

Our research combines work on time series summarization, management and pattern mining, and work on spatio-temporal databases for handling sensor data for traffic management. Many dimensionality reduction methods for time series have been proposed [9,10], e.g., to mine for patterns. Approximation methods can be categorized into two main families - functions that transform sequences into segments, and those that transform the initial dataset. The first category includes for instance methods PAA, or TIDES [11]. Transformation methods map the time series into some function space, e.g., Discrete Fourier Transform, or Wavelet Transformations. Symbolic representation allows representing a series by a sequence of discrete symbols, that can be applied for instance to each line segment [7], or to angles between lines [11]. STPCA belongs to the first family.

Spatio-temporal database research on traffic networks involves many issues. In many cases, the network is transformed into a (spatial) graph, which is used as a basis for planning maintenance and expansion of the network. Hence, operations research and graph theory play important roles in this activity – e.g., [12] is concerned with routing models, using computation of flow in the network graph. The work of [13] also takes advantage of network graphs and graph properties to optimize kNN and distance range queries in traffic. Graphs and network structures are used to forecast and analyze trajectories of mobile objects (e.g.,[14] in indexing trajectories). There is also extensive research on models for mobile objects and trajectories – e.g., [15,16]. These models assume that the identity of each object is known, and that the sensors are placed on the moving objects.

## 6 Conclusions and Ongoing Work

This paper presented research conducted within a multidisciplinary project, in the domain of spatio-temporal sensor data processing for urban traffic. It combines research on analytical methods to pre-process, clean and summarize multiple sensor data sources, and research on spatio-temporal database management. Introduction of STPCA improves data analysis considering the spatio temporal aspect of the dataset, offering a compression factor which outperforms those obtain by classical S-mode PCA and T-mode PCA with an equivalent reconstruction error. STPCA also dominates both other methods to estimate missing values and atypical cases. As part of the work in CADDY, we have constructed a prototype that allows the visual exploration of sensor data, shown in figure 5.

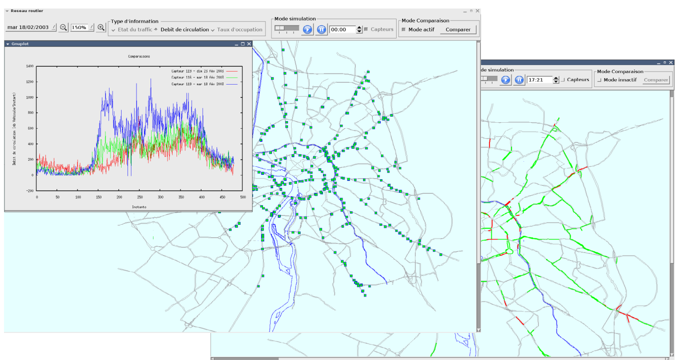


**Fig. 5.** Screen copy of prototype

Future work involves theoretical and implementation issues. The latter mainly concern developing more tools within the prototype to directly support the classes of queries described. This will require, among others, linking the prototype to pattern recognition algorithms. We furthermore need to explore other extensions to STPCA. This method is geared towards studying individual sensors, for specific patterns, to determine an average typical behavior for each sensor. STPCA can be extended to spatial aggregation (several sensors) over an area. Another issue concerns deriving other kinds of traffic behavior description – e.g., for atypical days, due to variation in human activity in a given area, such as holidays or festivals. A combination of these two extensions would support a wider variety of traffic pattern descriptions – e.g., for distinct events. This, in turn, would allow new kinds of decision support in real time traffic management, including interaction with intelligent car systems.

**Acknowledgements.** This work was partially financed by CNPq (Brazil) and by the French Research Program "ACI Masse de Données 2003".

## References

1. CADDY: The CADDY Website - (2007), http://norma.mas.ecp.fr/wikimas/Caddy
2. Scemama, G., Carles, O.: Claire-SITI, Public road Transport Network Management Control: a Unified Approach. In: 12th IEEE Int. Conf. on Road Transport Information and Control (RTIC 2004) (2004)
3. Joliveau, M.: Reduction of Urban Traffic Time Series from Georeferenced Sensors, and extraction of Spatio-temporal series - in French. Ph.D thesis, Ecole Centrale Des Arts Et Manufactures (Ecole Centrale de Paris) (2008)
4. Jolliffe, I.: Principal Component Analysis. Springer, New York (1986)
5. Joliveau, M., Vuyst, F.D.: Space-time summarization of multisensor time series. case of missing data. In: Int. Workshop on Spatial and Spatio-temporal data mining, IEEE SSTDM (2007)
6. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood for incomplete data via the em algorithm. Journal of the Royal Statistical Society series B 39, 1–38 (1977)
7. Hugueney, B.: Adaptive Segmentation-Based Symbolic Representations of Time Series for Better Modeling and Lower Bounding Distance Measures. In: Proc. 10th European Conference on Principles and Practice of Knowledge Discovery in Databases, pp. 542–552 (2006)
8. Hugueney, B., Joliveau, M., Jomier, G., Manouvrier, M., Naja, Y., Scemama, G., Steffan, L.: Towards a data warehouse for urban traffic (in french). Revue des Nouvelles Technologies de L'Information RNTI (B2), 119–137 (2006)
9. Yi, B.K., Faloutsos, C.: Fast time sequence indexing for arbitrary Lp norm. In: Proc. of the 26th VLBD Conference, pp. 385–394 (2000)
10. Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S.: Dimensionality reduction for fast similarity search in large time series databases. Journal of Knowledge and Information Systems (2000)
11. Mariotte, L., Medeiros, C.B., Torres, R.: Diagnosing Similarity of Oscillation Trends in Time Series. In: International Workshop on spatial and spatio-temporal data mining - SSTDM, pp. 243–248 (2007)
12. Mautora, T., Naudin, E.: Arcs-states models for the vehicle routing problem with time windows and related problems. Computers and Operations Research 34, 1061–1084 (2007)
13. Kriegel, H.P., Kröger, P., Kunath, P., Renz, M., Schmidt, T.: Proximity queries in large traffic networks. In: Proc. ACM GIS, pp. 1–8 (2007)
14. Kim, K., Lopez, M., Leutenegger, S., Li, K.: A Network-based Indexing Method for Trajectories of Moving Objects. In: Yakhno, T., Neuhold, E.J. (eds.) ADVIS 2006. LNCS, vol. 4243, pp. 344–353. Springer, Heidelberg (2006)
15. Guting, R., Bohlen, M., Erwig, E., Jensen, C., Lorentzos, N., Schneider, M., Vazirgianis, M.: A Foundation for Representing and Querying Moving Objects. ACM Transactions on Database Systems 25(2), 1–42 (2000)
16. Spaccapietra, S., Parent, C., Damiani, M.L., Macedo, J.A., Porto, F., Vangenot, C.: A conceptual view on trajectories. Knowledge and Data Engineering 65(1), 126–146 (2008)

# Retrieving Documents with Geographic References Using a Spatial Index Structure Based on Ontologies*

Miguel R. Luaces, Ángeles S. Places, Francisco J. Rodríguez, and Diego Seco

Databases Laboratory, University of A Coruña
Campus de Elviña, 15071 A Coruña, Spain
{luaces,asplaces,franjrm,dseco}@udc.es

**Abstract.** Both *Geographic Information Systems* and *Information Retrieval* have been very active research fields in the last decades. Lately, a new research field called *Geographic Information Retrieval* has appeared from the intersection of these two fields. The main goal of this field is to define index structures and techniques to efficiently store and retrieve documents using both the text and the geographic references contained within the text.

We present in this paper a new index structure that combines an inverted index, a spatial index, and an ontology-based structure. This structure improves the query capabilities of other proposals. In addition, we describe the architecture of a system for geographic information retrieval that uses this new index structure. This architecture defines a workflow for the extraction of the geographic references in the document.

## 1 Introduction

Two research fields that have received much attention during the last years are those of Information Retrieval [1] and Geographic Information Systems [2]. These fields have produced industry product lines such as digital libraries, document databases, web search engines, and spatial data infrastructures [3]. During the last decades these two research fields have advanced independently. Although it is very common that textual and geographic information occur together in information systems, the geographic references of documents are rarely used in information retrieval systems. Few index structures or retrieval algorithms take into account the spatial nature of geographic references embedded within documents. Pure textual techniques focus only on the language aspects of the documents and pure spatial techniques focus only on the geographic aspects of the documents. None of them are suitable for a combined approach to information retrieval because they completely neglect the other type of information.

* This work has been partially supported by "Ministerio de Educación y Ciencia" (PGE y FEDER) ref. TIN2006-16071-C03-03, by "Xunta de Galicia" ref. PGIDIT05SIN10502PR and ref. 2006/4, and by "Consellería de Educación e Ordenación Universitaria da Xunta de Galicia" for Diego Seco.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 395–404, 2008.

As a result, there is a lack of system architectures, index structures and query languages that combine both types of information.

Some proposals have appeared recently [4,5,6] that define new index structures that take into account both the textual and the geographic aspects of a document. These proposals are the origin of a new research field called Geographic Information Retrieval (GIR). However, there are some specific particularities of geographic space that are not taken into account by these approaches. Particularly, concepts such as the hierarchical nature of geographic space and the topological relationships between the geographic objects must be considered in order to fully represent the relationships between the documents and to allow new and interesting types of queries to be posed to the system.

In this paper, we present an index structure that takes these issues into account. We first describe some basic concepts and related work in Section 2. Section 3 describes our index structure and the procedures used to built it. Then, in Section 4, we present the general architecture of the system and describe its components. After that, in Section 5, we describe some types of queries that can be answered with this system and we sketch the algorithms that can be used to solve this queries. Furthermore, Section 6 presents some experiments that we made to compare our structure with other ones that use a pure spatial index. Finally, Section 7 presents some conclusions and future lines of work.

## 2 Related Work

Inverted indexes are considered the classical text indexing technique [1]. The main drawback of these indexes is that geographic references are mostly ignored because place names are considered words just like the others. If the user poses a query such as as *hotels in Spain*, the place name *Spain* is considered a word, and only those documents that contain exactly that word are retrieved. Regarding indexing geographic information, many different spatial index structures have been proposed along the years. A good survey of these structures can be found in [7]. A drawback of spatial index structures is that they do not take into consideration the geographic ontology of the real world. Internal nodes in the structure are meaningless in the real world and it is not possible to associate location-specific information to these nodes because there is no relation at all between the nodes in the spatial index structure and real world locations.

Some work has been done to combine both types of indexes. Finding geographical references in text is a very difficult problem and there have been many papers that deal with different aspects of this problem and describe complete systems such as Web-a-where [8], MetaCarta [9], and STEWARD [4]. The papers about the SPIRIT (Spatially-Aware Information Retrieval on the Internet) project [10,11,12,13] are a very good starting point. In [12], the authors conclude that keeping separate text and spatial indexes, instead of combining both in one, results in less storage costs but it could lead to higher response times. More recent works can be broadly classified into two categories depending on how they combine textual and spatial indexes. On the one hand, some proposals have appeared that combine textual and spatial aspects in an hybrid index [14,15]. On the other hand, other proposals define structures that keep separate indexes for spatial and text attributes [4,5,6]. Our index structure is part of this second group because this division has many advantages [6]. Nevertheless, none of these approaches take into account the relationships between the geographic objects that they are indexing.

A structure that can properly describe the specific characteristic of geographic space is an *ontology*, which is a formal explicit specification of a shared conceptualization [16,17]. An ontology provides a vocabulary of classes and relations to describe a given scope. In [18], a method is proposed for the efficient management of large spatial ontologies using a spatial index to improve the efficiency of the spatial queries. Furthermore, in [10,13] the authors describe how ontologies are used in query term expansion, relevance ranking, and web resource annotation in the SPIRIT project. However, as far as we know, nobody has ever tried to combine ontologies with other types of indexes to have a hybrid structure that captures both the spatial and the semantic relationships between the geographic objects indexed.

## 3 Index Structure

Our index architecture has three main components, a textual index, a spatial index, and a place name hash table to optimize the resolution of a particular type of very common queries. The textual index is built using Lucene [19] by parsing and inserting each of the documents into the index. The *place name hash table* stores for each location name its position in the spatial index structure. This provides direct access to a single node by means of a keyword that is returned by the Geographic Space Ontology Service (see Figure 2) if the word processed is a location name.

The spatial index is based on an ontology [16,17] of the geographic space that describes the concepts in our domain and the relationships that hold between them. There are different ontology languages that provide different formal and reasoning facilities. OWL [20] is a W3C standard language to describe ontologies and can be categorised into three species or sub-languages: OWL-Lite, OWL-DL and OWL-Full. Our spatial ontology is described in OWL-DL and it can be downloaded from the following URL: `http://lbd.udc.es/ontologies/spatialrelations`.

OWL classes can be interpreted as sets that contain individuals (also known as instances). Individuals can be referred to as being *instances of classes*. Our ontology describes eight classes of interest: *SpatialThing*, *GeographicalThing*, *GeographicalRegion*, *GeopoliticalEntity*, *PopulatedPlace*, *Region*, *Country*, and *Continent*. In our ontology there are hierarchical relations among *SpatialThing*, *GeographicalThing*, *GeographicalRegion*, *GeopoliticalEntity* because:

- *GeopoliticalEntity* is subclass of *GeographicalRegion*
- *GeographicalRegion* is subclass of *GeopoliticalEntity*
- *GeopoliticalEntity* is subclass of *GeographicalThing* and
- *GeographicalThing* is subclass of *SpatialThing*.

That is, these four classes are organised into a superclass-subclass hierarchy, which is also known as a taxonomy. Subclasses specialise (are subsumed by) their superclasses. *GeopoliticalEntity* has four subclasses: *PopulatedPlace*, *Country*, *Continent*, and *Region*. All the individuals are members of these subclasses. These four subclasses have an additional necessarily asserted condition regarding their relations with each other. They are connected by the property *spatiallyContainedBy* that describes the existence of an spatial relationship among them. For instance, all the individuals of class *PopulatedPlace* are *spatiallyContainedBy* individuals of class *Region* (described in OWL as *PopulatedPlace spatiallyContainedBy only (AllValuesFrom) Region*). Figure 1 shows an example of these relationships. Ontology classes are represented as circles, individuals as rectangles, and the relationships as labelled lines.

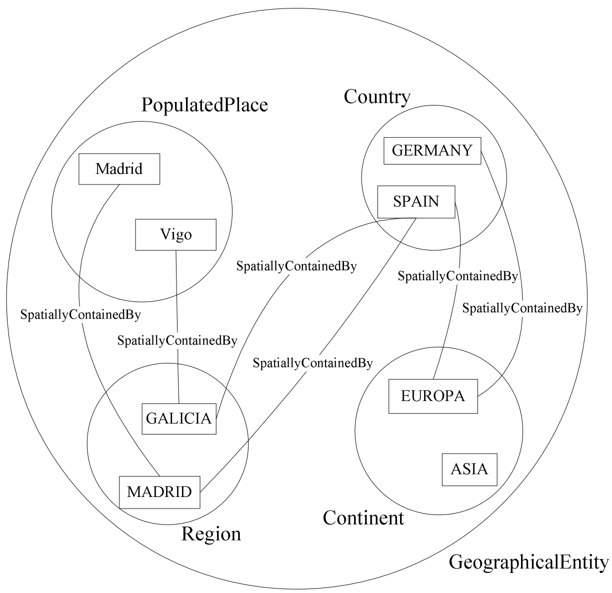



**Fig. 1.** Ontology instances

We build the ontology using a *Gazetteer* (in our test implementation we use *Geonames* [21]). However, Geonames (and Gazetteers in general) does not provide geometries for the location names other than a single representative point whereas our spatial index needs the real geometry of the location name (for example, the boundary of countries). Hence, we defined a *Geometry Supplier* service to obtain the geometries of those location names. As a base for this service we used the *Vector Map* (VMap) cartography [22]. The Geographic Space Ontology Service is composed of both the Gazetteer service and the Geometry Supplier service.

The spatial indexing stage comprises three steps. First, the system extracts *candidate location names* (words that are likely to be location names) from the text. The documents are parsed in order to discover the place names contained within. We use the *Natural Language Tool LingPipe* [23] to find the candidate location names. In the system prototype we use LingPipe trained with the MUC6 corpus (http://www.ldc.upenn.edu) labelled with locations, people and organizations. After the LingPipe processing, the module filters the resultant named entities selecting only the locations and discarding people and organization names.

In a second step, the candidate locations are processed in order to determine whether the candidates are real location names, and, in this case, to compute their geographic locations. There are some problems that can happen at this point: a location name can be ambiguous (*polysemy*), and there can be multiple names for the same geographic location. We developed a module, called Geographic Space Ontology Service, based on the ontology of the geographic space to geo-reference location names. This module returns for a candidate location name an *ontology graph* with the individual that represents the location name and all the individuals related by means of *spatiallyContainedBy* relationships. If the ontology does not have an individual for the candidate location name, it is discarded.

Finally, the third step consists in building the spatial index with the ontology graphs of the geo-referenced locations computed in the previous step together with references to the documents containing them. The spatial index is a tree composed by nodes that represent location names. The tree structure depends on the ontology that is used in the system. In the case of the ontology described previously, the nodes are connected by means of inclusion relationships (for instance, *Galicia* is included in *Spain*). In each node we store: (i) the keyword (a place name), (ii) the bounding box of the geometry representing this place, (iii) a list with the document identifiers of the documents that include geographic references to this place, and (iv) a list of child nodes that are geographically within this node. If the list of child nodes is very long, using sequential access is very inefficient. For this reason, if the number of children nodes exceeds a threshold, an R-Tree is used instead of a list.

This structure has two main drawbacks. First, the tree that supports the structure is possibly unbalanced penalizing the efficiency of the system. We present some experiments in Section 6 trying to prove that this is not a very important problem. Second, ontological systems have a fixed structure and thus our structure is static and it must be constructed *ad-hoc*.

## 4 System Architecture

Figure 2 shows our proposal for the system architecture of a geographic information retrieval system. The architecture can be divided into three independent layers: the index construction workflow, the processing services, and the user interfaces. The bottom part of the figure shows the index construction workflow, which, in turn, consists of three modules: the document abstraction module, the index construction module, and the index structure itself.

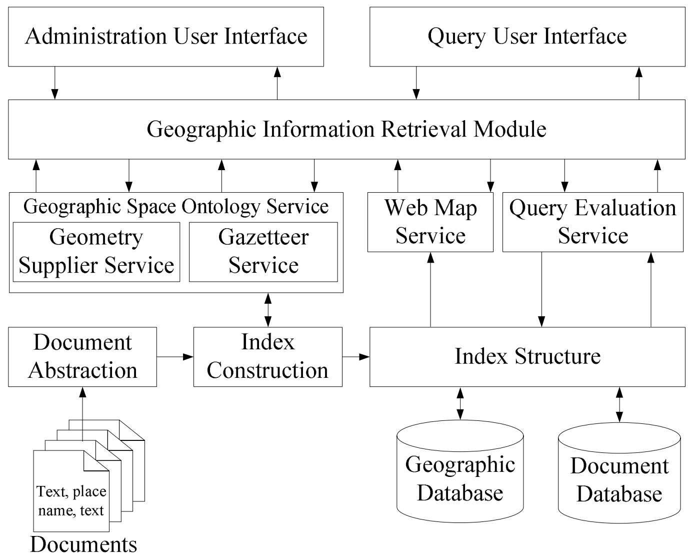



**Fig. 2.** System Architecture

The processing services are shown in the middle of the figure. On the left side, the *Geographic Space Ontology Service* used in the spatial index construction is shown. On the right side, one can see the two services that are used to solve queries. The rightmost one is the *query evaluation service*, which receives queries and uses the index structure to solve them. Section 5 describes the types of queries that can be solved by this service, as well as the algorithms that are used to solve these queries. The other service is a *Web Map Service* following the OGC specification [24] that is used to create cartographic representations of the query results. On top of these services a *Geographic Information Retrieval Module* is in charge of coordinating the task performed by each service to response the user requests.

The topmost layer shows the two user interfaces that exist in the architecture: the *Administration User Interface* and the *Query User Interface*. The administration user interface was developed as a stand-alone application and it can be used to manage the document collection. The *Query User Interface* interface was developed as a web application using the *Google Maps API* [25]. This user interface allows the user to indicate both the textual and the spatial aspects of queries. The spatial context can be introduced in three ways that are mutually exclusive: typing the location name, selecting the location name in a tree, and visualizing the spatial context of interest in the map.

## 5 Supported Query Types

The most important characteristic of an index structure is the type of queries that can be solved with it. Our index structure support three types of queries:

pure textual queries, pure spatial queries, and queries with a textual and a spatial component. In this last type, the spatial component can be given both as a location name or as a geographical area.

Pure textual queries such as "*retrieve all documents where the words* hotel *and* sea *appear*" can be solved by our system because a textual index is part of the index structure. Similarly, pure spatial queries such "*retrieve all documents that refer to the following geographic area*" can also be solved because the index structure is built like a spatial index. Each node in the tree is associated with the bounding box of the geographic objects in its subtree. Therefore, the same algorithm that is used with spatial indexes can be used with our structure.

Furthermore, the index structure that we propose can be used to solve queries that involve a textual and a spatial component. In this case, the textual index is used to retrieve the list of documents that contain the words, and the spatial index structure is used to compute the list of documents that reference the geographic area. The result to the query is computed as the intersection of both lists. In the case of queries such as "*retrieve all documents with the word* hotel *that refer to Spain*", our system uses the Geographic Space Ontology Service to discover that *Spain* is a geographic reference and then it uses the *place name hash table* to retrieve the index node that represents *Spain*. Thus, we save some time by avoiding a tree traversal.

Another improvement over text and spatial indexes is that our index structure can easily perform query expansion on geographic references because the index structure is built from an ontology of the geographic space. Consider the following query "*retrieve all documents that refer to Spain*". The query evaluation service will discover that Spain is a geographic reference and the place name index will be used to quickly locate the internal node that represents the geographic object *Spain*. Then all the documents associated to this node are part of the result to the query. Moreover, all the children of this node are geographic objects that are contained within Spain (for instance, the city of Madrid). Therefore, all the documents referenced by the subtree are also part of the result of the query. The consequence is that the index structure has been used to expand the query because the result contains not only those documents that include the term *Spain*, but also all the documents that contain the name of a geographic object included in Spain (e.g., all the cities and regions of Spain).

## 6 Experiments

In the previous section we showed that our structure has a qualitative advantage over systems that combine a textual index with a pure spatial index because query expansion can be performed directly with our index structure. Hence, our index structure supports a new type of query that cannot be implemented with a pure spatial index. However, unlike pure spatial index structures, our index structure is not balanced and therefore, the query performance can be worse. In this section we describe the experiments that we performed to compare our structure with other ones based on a pure spatial index. We used the TREC FT-91 (Financial Times, year 1991) document collection [26], which consists of 5,368 news documents. Then, we built two indexes over this collection: one using our index structure as described in this paper, and another one using a textual index and an R-Tree. Furthermore, we developed an algorithm to generate random spatial query windows based on the performance comparisons of the R*-Tree in [27]. We compared the structures with respect to four different query window areas, namely 0.001%, 0.01%, 0.1% and 1% of the world. We generated 100,000 random query windows for each area, and we averaged the computing time of each query execution. Table 1 shows the results of this experiment.

**Table 1.** Ontology-based index versus R-Tree

| | **Overall** | | | | **High density** | | | | **Low density** | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Query area (%)** | 0.001 | 0.01 | 0.1 | 1 | 0.001 | 0.01 | 0.1 | 1 | 0.001 | 0.01 | 0.1 | 1 |
| **Our index** | 0.013 | 0.017 | 0.052 | 0.360 | 0.03 | 0.11 | 1.05 | 9.84 | 0.02 | 0.03 | 0.09 | 0.4 |
| **R-Tree** | 0.010 | 0.016 | 0.057 | 0.370 | 0.07 | 0.22 | 1.64 | 12.85 | 0.02 | 0.03 | 0.07 | 0.2 |

The first row of the table shows the results obtained with our structure (in milliseconds), and the second one shows the results obtained with the structure using an R-Tree. Both index structures have similar performance. The performance of our structure is a bit worse than the R-Tree when the query window is small but, surprisingly it is a bit better than the R-Tree when the query window is bigger. In order to explain this surprising result, we analyzed the performance in particular zones. We distinguished two relevant types of zones and we repeated the experiment generating random queries in both zones.

First, we studied the performance of the structures when the document density is high. In this case, the performance of our structure is higher than the R-Tree performance. We believe this is because our structure stores a list of documents for each location while the R-Tree uses a node for each document.

Then, we studied the performance when the document density is low. In this case, the R-Tree performance is better because the number of nodes in both structures is similar and the R-Tree is balanced whereas our structure may be unbalanced. For this reason, in the general case, when the query window is small the probability of that query window being in a high document density zone is small and, therefore, the R-Tree performance is better. However, when the query window is bigger that probability is higher and, therefore, the R-Tree performance is lower.

## 7 Conclusions and Future Work

We have presented in this paper a system architecture for an information retrieval system that takes into account not only the text in the documents but also the geographic references included in the documents and the ontology of the geographic space. This is achieved by a new index structure that combines a textual index, a spatial index, and an ontology-based structure. We have also presented how traditional queries can be solved using the index structure, and new types of queries that can be solved with the index structure are described and the algorithms that solve these queries are sketched. Finally, we performed some experiments that show that the performance of our structure is acceptable in comparison with index structures using pure spatial indexes.

Future improvements of this index structure are possible. We are currently working on the evaluation of the performance of the index structure, particularly we are performing experiments to determine the precision and recall. Moreover, *Toponym Resolution* techniques must be implemented to solve ambiguity problems when we geo-reference the documents. Another line of future work involves exploring the use of different ontologies and determining how each ontology affects the resulting index. Furthermore, we plan on including other types of spatial relationships in the index structure in addition to inclusion (e.g., adjacency). These relationships can be easily represented in the ontology-based structure, and the index structure can be extended to support them. Finally, it is necessary to define algorithms to rank the documents retrieved by the system. For this task, we must define a measure of spatial relevance and combine it with the relevance computed using the inverted index.

## References


1. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison Wesley, Reading (1999)
2. Worboys, M.F.: GIS: A Computing Perspective. CRC, Boca Raton (2004)
3. Global Spatial Data Infrastructure Association: Online documentation (Retrieved, March 2008), http://www.gsdi.org/
4. Lieberman, M.D., Samet, H., Sankaranarayanan, J., Sperling, J.: STEWARD: Architecture of a Spatio-Textual Search Engine. In: Proceedings of the 15th ACM Int. Symp. on Advances in GIS (ACMGIS 2007), pp. 186–193. ACM Press, New York (2007)
5. Chen, Y.Y., Suel, T., Markowetz, A.: Efficient query processing in geographic web search engines. In: SIGMOD Conference, pp. 277–288 (2006)
6. Martins, B., Silva, M.J., Andrade, L.: Indexing and ranking in Geo-IR systems. In: GIR 2005: Proceedings of the 2005 workshop on Geogr. Inform. retrieval, pp. 31–34. ACM Press, New York (2005)
7. Gaede, V., Günther, O.: Multidimensional access methods. ACM Comput. Surv. 30(2), 170–231 (1998)
8. Amitay, E., Har'El, N., Sivan, R., Soffer, A.: Web-a-where: geotagging web content. In: SIGIR 2004: Proceedings of the 27th ACM SIGIR, pp. 273–280. ACM, New York (2004)
9. Rauch, E., Bukatin, M., Baker, K.: A confidence-based framework for disambiguating geographic terms. In: Proceedings of the HLT-NAACL 2003 workshop on Analysis of Geogr. references, Morristown, USA, pp. 50–54. Association for Computational Linguistics (2003)

10. Jones, C.B., Abdelmoty, A.I., Fu, G.: Maintaining ontologies for geographical information retrieval on the web. In: Meersman, R., Tari, Z., Schmidt, D.C. (eds.) CoopIS 2003, DOA 2003, and ODBASE 2003. LNCS, vol. 2888, pp. 934–951. Springer, Heidelberg (2003)
11. Jones, C.B., Abdelmoty, A.I., Fu, G., Vaid, S.: The SPIRIT Spatial Search Engine: Architecture, Ontologies and Spatial Indexing. In: Egenhofer, M.J., Freksa, C., Miller, H.J. (eds.) GIScience 2004. LNCS, vol. 3234, pp. 125–139. Springer, Heidelberg (2004)
12. Vaid, S., Jones, C.B., Joho, H., Sanderson, M.: Spatio-Textual Indexing for Geographical Search on the Web. In: Bauzer Medeiros, C., Egenhofer, M.J., Bertino, E. (eds.) SSTD 2005. LNCS, vol. 3633, pp. 19–36. Springer, Heidelberg (2005)
13. Fu, G., Jones, C.B., Abdelmoty, A.I.: Ontology-Based Spatial Query Expansion in Information Retrieval. In: Meersman, R., Tari, Z. (eds.) OTM 2005. LNCS, vol. 3761, pp. 1466–1482. Springer, Heidelberg (2005)
14. Zhou, Y., Xie, X., Wang, C., Gong, Y., Ma, W.Y.: Hybrid index structures for location-based web search. In: Proceedings of CIKM 2005, pp. 155–162. ACM, New York (2005)
15. Hariharan, R., Hore, B., Li, C., Mehrotra, S.: Processing Spatial-Keyword (SK) Queries in Geographic Information Retrieval (GIR) Systems. In: Proceedings of the 19th Int. Conf. on Scientific and Statistical Database Management (SSDBM 2007). IEEE Computer Society, Los Alamitos (2007)
16. Gruber, T.R.: A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition 5(2), 199–220 (1993)
17. Gruber, T.R.: Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In: Guarino, N., Poli, R. (eds.) Formal Ontology in Conceptual Analysis and Knowledge Representation, Deventer, The Netherlands. Kluwer Academic Publishers, Dordrecht (1993)
18. Dellis, E., Paliouras, G.: Management of Large Spatial Ontology Bases. In: Proceedings of the Workshop on Ontologies-based techniques for DataBases and Information Systems (ODBIS) of the 32nd Int. Conf. on Very Large Data Bases (VLDB 2006) (September 2006)
19. Apache: Lucene (Retrieved, March 2008), http://lucene.apache.org
20. World Wide Consortium: Owl web ontology language reference (Retrieved March, 2008), http://www.w3.org/TR/owl-ref
21. Geonames: Gazetteer (Retrieved, March 2008), http://www.geonames.org
22. National Imagery and Mapping Agency (NIMA): Vector Map Level 0 (Retrieved, March 2008), http://www.mapability.com
23. Alias-i: LingPipe, Natural Language Tool (Retrieved, March 2008), http://www.alias-i.com/lingpipe/
24. Open GIS Consortium, Inc.: OpenGIS Web Map Service Implementation Specification. OpenGIS Project Document 01-068r3, Open GIS Consortium, Inc. (2002)
25. Google: Google Maps API (Retrieved, March 2008), http://code.google.com/apis/maps/
26. National Institute of Standards and Technology (NIST): TREC Special Database 22, TREC Document Database: Disk 4 (Retrieved, March 2008), http://www.nist.gov/srd/nistsd22.htm
27. Beckmann, N., Kriegel, H.P., Schneider, R., Seeger, B.: The R*-tree: an efficient and robust access method for points and rectangles. SIGMOD Rec. 19(2), 322–331 (1990)

# Preface to WISM 2008

Flavius Frasincar[1], Geert-Jan Houben[2], and Philippe Thiran[3]

[1] Erasmus University Rotterdam, The Netherlands
[2] Vrije Universiteit Brussel & TU Eindhoven, Belgium & The Netherlands
[3] Namur University, Belgium

The international workshop on Web Information Systems Modeling (WISM) aims at covering the latest developments for model-driven design of Web Information Systems (WIS). This is the fifth edition of the workshop, after four successful editions organized in Trondheim (2007), Luxembourg (2006), Sydney (2005), and Riga (2004).

The first paper, the invited paper, by Ma et al. discusses Abstract State Services as an abstraction of Web services. Abstract State Services use a hidden database layer combined with a function-enabled view layer. These abstractions enable easy integration and composition of existing WIS components for building new WIS.

The second paper by De Virgilio and Torlone proposes the General Hypertext Model, a reference meta-model for WIS navigational design. The proposed hypertext model generalizes both data-oriented and object-oriented approaches for navigational design. The authors discuss three scenarios that make use of General Hypertext Model: hypertext exchange, hypertext integration, and hypertext adaptation.

The third paper by Luinenburg et al. applies situational engineering to the development of WIS. The approach is based on identifying the key features for WIS design and subsequently selecting the ones required by a WIS. After that, fragments of possibly different design methods that support these features are selected and assembled in a specific design method for the desired WIS.

The fourth paper by Alpuente et al. presents a rule-based specification and verification language for WIS. The language makes use of an ontology which allows the use of semantic information during rule expansion. The integration between the rule-language and the ontology is achieved by extending OWL-DL with variables and function calls.

The fifth paper by Rigo and de Oliveira proposes the acquisition of user stereotypes by using Web usage mining and domain ontologies. The access patterns have associated semantic contexts that describe the semantic relations between pages as well as page content similarity. Based on the semantic context and user current information, the system is adapted by suggesting links and topics of possible interest to the user.

The sixth paper and last one in this workshop by Milea et al. aims at supporting WIS that make use of Semantic Web technologies for modeling the temporal context. The paper proposes TOWL, an extension of OWL-DL with temporal constructs based on concrete domains and fluents. One of these extensions refers to temporal cardinality that lifts the OWL-DL cardinality restrictions in a temporal dimension.

We do hope that the topics of the selected papers will appeal to the reader and invite her/him to have a closer look at the articles gathered in the proceedings. Last, but not least, we would like to thank all the authors, reviewers, and participants for their contributions and support making thus the organization of the workshop possible.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, p. 405, 2008.

# Abstract State Services

## A Theory of Web Services

Hui Ma[1], Klaus-Dieter Schewe[2], Bernhard Thalheim[3], and Qing Wang[4]

[1] Victoria University of Wellington, School of Mathematics, Statistics and Computer Science, Wellington, New Zealand
hui.ma@mcs.vuw.ac.nz
[2] Information Science Research Centre, Palmerston North, New Zealand
kdschewe@acm.org, isrc@xtra.co.nz
[3] Christian-Albrechts-University Kiel, Institute of Computer Science, Kiel, Germany
thalheim@is.informatik.uni-kiel.de
[4] Massey University, Palmerston North, New Zealand
q.q.wang@massey.ac.nz

**Abstract.** Abstract State Services (ASSs) have been introduced recently as an abstraction of web services that exploit the fundamental approach of Abstract State Machines. An ASS combines a hidden database layer with an operation-equipped view layer, and can be a simple function, a data warehouse or a full-fledged Web Information System (WIS). In this paper we provide a language for ASSs, and show how ASSs capture all these instantiations of "services".

## 1 Introduction

Since its introduction the role of the world-wide web has shifted from enabling access to a pool of documents to the provision of services. Such web services can in fact be anything from a simple function to a fully functional Web Information System (WIS). The unifying characteristic is that content and functionality are made available for use by human users or other services. Therefore, web service integration has become a highly relevant research topic, and a lot of work has been investigated into it [1]. However, despite the existence of conference series on web services and a lot of buzz around "service-oriented architecture" (SOA) the notion of "service" has surprisingly been defined only vaguely.

In this paper we are particularly interested in an abstract, conceptual approach to service integration and composition that has recently been introduced by means of postulates for Abstract State Services (ASSs) [2]. ASSs adopt the fundamental idea from WISs that a service can be described by two layers: a hidden database layer consisting of a database schema and transactions, and a visible view layer on top of it providing views and functions based on them [3]. The postulates for ASSs follow the line of thought of the ASM thesis [4], for which Blass and Gurevich formalised sequential and parallel algorithms by requiring a small set of intuitive, abstract postulates, and proved

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 406–415, 2008.

that these postulates are captured by (sequential) Abstract State Machines (ASMs) [5].

We show that ASSs capture services as diverse as simple functions, data warehouses and WISs. In particular, our focus is on describing WISs in general as particular ASSs. For this we pick up the concept of Abstract Database Transformation Machines (ADTMs), a modification of ASMs that captures the postulates for database transformations [6], and enhance ADTMs by integrating them with a declarative query language that is based on fixed-point construction and identifier creation.

Regarding WISs as ASSs allows us to benefit from the approach to ASS integration and composition that was developed in [2]. In this way we enable the construction of WISs by means of plugging in components from various existing WISs, i.e. we achieve WIS interoperability. Furthermore, as the concept of ASS is not bound to WISs, we may even be able to integrate WISs with any kind of web service. In particular, the close relationship between ADTMs and ASMs permits the easy integration of web services that have been specified by means of ASMs [7].

The remainder of this paper is organised as follows. In Section 2 we briefly introduce the model of Abstract State Services without detailed reference to the postulates for database transformations. We discuss how functional web services, data warehouses and WISs can be seen as a special case of ASSs. In Section 3 we develop an abstract language for ASSs based on the results in [6] and the media type concept for WISs from [3]. We demonstrate the language by a concise WIS example. We conclude with a brief summary and discussion of future research.

## 2 Abstract State Services

Following the general approach of Abstract State Machines [4] we may consider each database computation as a sequence of abstract states, each of which represents the database (instance) at a certain point in time plus maybe additional data that is necessary for the computation, e.g. transaction tables, log files, etc. In order to capture the semantics of transactions we distinguish between a wide-step transition relation and small step transition relations. A transition in the former one marks the execution of a transaction, so the wide-step transition relation defines infinite sequences of transactions. Without loss of generality we can assume a serial execution, while of course interleaving is used for the implementation. Then each transaction itself corresponds to a finite sequence of states resulting from a small step transition relation, which should then be subject to the postulates for database transformations [6].

A *database system* DBS consists of a set $\mathcal{S}$ of states, together with a subset $\mathcal{I} \subseteq \mathcal{S}$ of initial states, a wide-step transition relation $\tau \subseteq \mathcal{S} \times \mathcal{S}$, and a set $\mathcal{T}$ of transactions, each of which is associated with a small-step transition relation $\tau_t \subseteq \mathcal{S} \times \mathcal{S}$ $(t \in \mathcal{T})$ satisfying the postulates of a database transformation over $\mathcal{S}$.

A *run* of a database system DBS is an infinite sequence $S_0, S_1, \dots$ of states $S_i \in \mathcal{S}$ starting with an initial state $S_0 \in \mathcal{I}$ such that for all $i \in \mathbb{N}$ $(S_i, S_{i+1}) \in \tau$ holds, and there is a transaction $t_i \in \mathcal{T}$ with a finite run $S_i = S_i^0, \dots, S_i^k = S_{i+1}$ such that $(S_i^j, S_i^{j+1}) \in \tau_{t_i}$ holds for all $j = 0, \dots, k-1$.

Views in general are expressed by queries, i.e. read-only database transformations. Therefore, we can assume that a view on a database state $S_i \in \mathcal{S}$ is given by a finite run $S_i = S_0^v, \dots, S_\ell^v$ of some database transformation $v$ with $S_i \subseteq S_\ell^v$ – traditionally, we would consider $S_\ell^v - S_i$ as the view. We can use this to extend a database system by views.

In doing so we let each state $S \in \mathcal{S}$ to be composed as a union $S_d \cup V_1 \cup \dots \cup V_k$ such that each $S_d \cup V_j$ is a view on $S_d$. As a consequence, each wide-step state transition becomes a parallel composition of a transaction and an operation that switches views on and off. This leads to the definition of an Abstract State Service (ASS).

An *Abstract State Service* (ASS) consists of a database system DBS, in which each state $S \in \mathcal{S}$ is a finite composition $S_d \cup V_1 \cup \dots \cup V_k$, and a finite set $\mathcal{V}$ of (extended) views. Each view $v \in \mathcal{V}$ is associated with a database transformation such that for each state $S \in \mathcal{S}$ there are views $v_1, \dots, v_k \in \mathcal{V}$ with finite runs $S_d = S_0^j, \dots, S_{n_j}^j = S_d \cup V_j$ of $v_j$ $(j = 1, \dots, k)$. Each view $v \in \mathcal{V}$ is further associated with a finite set $\mathcal{O}_v$ of (service) operations $o_1, \dots, o_n$ such that for each $i \in \{1, \dots, n\}$ and each $S \in \mathcal{S}$ there is a unique state $S' \in \mathcal{S}$ with $(S, S') \in \tau$. Furthermore, if $S = S_d \cup V_1 \cup \dots \cup V_k$ with $V_i$ defined by $v_i$ and $o$ is an operation associated with $v_k$, then $S' = S'_d \cup V'_1 \cup \dots \cup V'_m$ with $m \geq k-1$, and $V'_i$ for $1 \leq i \leq k-1$ is still defined by $v_i$.

In a nutshell, in an ASS we have view-extended database states, and each service operation associated with a view induces a transaction on the database, and may change or delete the view it is associated with, and even activate other views. These service operations are actually what is exported from the database system to be used by other systems or directly by users.

A formalisation of database transformations by means of postulates is beyond the scope of this paper and excluded due to space limitations. In a nutshell, the postulates require a one-step transition relation between states (sequential time postulate), states as (meta-finite) first-order structures (abstract state postulate), necessary background for database computations such as complex value constructors (background postulate), limitations to the number of accessed terms in each step (exploration boundary postulate), and the preservation of equivalent substructures in one successor state (genericity postulate) [2]. In [6] the language of Abstract Database Transformation Machines (ADTMs) has been developed, which captures all database transformations.

**Examples.** Let us now look at examples for ASSs. We will concentrate on functions, which are quite often taken as web services, Data Warehouses and Web Information Systems.

Functional Web Services. Suppose we have a database with employee information, in particular salaries. Individual salaries will be kept hidden, but building averages for groups for employees will be offered as a service. In this case, we could have a quaternary relation with employee_id, name, department and salary in the database schema. Using ASMs [5] we would model this by a controlled 4-ary function employee. Then employee(43,Lisa,Cheese,4100)=1 means that there is an employee with id 43, name Lisa, and salary 4100 in the Cheese department, while employee(552,Bernd,Milk,8000)=0 means that in the Milk department there is no employee named Bernd with id 552 and salary 8000.

The averaging operation would be made available in combination with an empty view. We would allow either a grouping by department or no grouping at all. The result would leave the database as it is but display a new view with the resulting relation and the same operation associated to it. Using ASM notation we could define the averaging operation by department simply by the rule

$$\text{result} := \{(d,a) \mid \exists i,n,s.\, \text{employee}(i,n,d,s) = 1 \wedge a = \text{avg}\langle s \mid \exists i,n,s.\, \text{employee}(i,n,d,s) = 1\rangle\}$$

Data Warehouses. A more interesting example of ASSs is given by data warehouses, which could be turned this way into web warehouses. The ASM-based approach in [8] used three linked ASMs to model data warehouse and OLAP applications. At its core we have an ASM modelling the data warehouse itself using star or snowflake schemata. For instance, here we could have controlled functions sales, product, and store all of arity 3, and a static ternary function time. As before, sales(003,14,27-2-2008)=1 represents the fact that product 003 was sold in store 14 on 27 February 2008, product(003,hammer,27.5)=1 means that the product with id 003 is a hammer, which is sold at a price of 27.5, store(14,Awapuni,Palmerston)=1 means that the store with id 14 is located in Awapuni in the city of Palmerston, and time(27,2,2008)=1 indicates that 27 February 2008 is a valid time point.

A second ASM would be used for modelling operational databases with rules extracting data from them and refreshing the data warehouse. This ASM is of no further relevance for us and thus will be ignored.

The third ASM models the OLAP interface on the basis of the idea that datamarts can be represented as extended views. Such a view may e.g. extract all sales in store 14 in 2008 together with product description and price. That is, the view defining database transformation could be described (using relational algebra operations liberally) by the simple rule

$$\text{result} := \pi_{\text{p-id},\text{description},\text{price},\text{day(date)},\text{month(date)}}(\sigma_{\text{s-id}=14 \wedge \text{year(date)}=2008}(\text{sales}) \bowtie \text{product})$$

Service operations associated with such a view could be roll-up and drill-down operations, e.g. aggregating sales per day or month, or slicing operations, e.g. concentrating on sales of a particular product. Furthermore, we could permit operations for changing the selected year or store. We omit further details.

Furthermore, the main rule in the OLAP ASM in [8] mainly serves the purpose of opening and closing datamarts and selecting operations associated with them. This has been already captured by the notion of a run.

Web Information Systems. Another even more complex example is given by Web Information Systems (WISs), following the modelling approach in [3], which among others provides the notion of media type. At its core, a media type is a view on a database schema that is extended by operations (and more), which is exactly what we capture with ASSs.

However, in this case the view-defining queries must be able to create the link structure between instances of media types, the so-called media objects, which implies that the creation of identifiers is a desirable property in such queries. As already stated the non-determinism in database transformation is motivated by such identifier creation. In this sense WISs provide an example for the necessity of non-determinisn in the small-step transition relations.

## 3 A Language for ASSs

In this section we develop a formal language for ASSs. Different from our work in [6] our emphasis is not so much completeness, but simply providing a language that is handy for capturing a wide class of WISs as ASSs.

**ADTMs.** As the definition of ASSs is mainly built on top of database transformations, we adopt the language of ADTMs from [6], which captures all database transformations. In the following we use the notation *atom(p)* referring to all atomic elements appeared in terms of a program $p$.

For a database transformation $t$ let $S$ be a state of $t$, $f$ a dynamic function symbol of arity $n$ in the state signature of $t$, and $a_1, ..., a_n, v$ be elements in the base set of $S$, then an *update* of $t$ is a pair $(l, v)$, where $l$ is a location $f(a_1, ..., a_n)$. An *update set* is a set of updates; an *update multiset* is a multiset of updates.

An update set $\mathcal{U} = \{(l_1, v_1), ..., (l_n, v_n)\}$ is *consistent* iff the locations of updates in $\mathcal{U}$ are distinct. With respect to a given state $S$, the execution of a non-trivial update set $\mathcal{U}$ over $S$ is denoted as $S \oplus \mathcal{U}$ such that $S \oplus \mathcal{U} = S'$ where $l_i^{S'} = v_i$ $(i = 1, ..., n)$ iff $\mathcal{U}$ is consistent, otherwise it is undefined.

An *Abstract Database Transformation Machine* (ADTM) $\Lambda$ has a fixed vocabulary $H$ over a finite set of base domains, consisting of a set $S_\Lambda$ of states of vocabulary $H$, a set $I_\Lambda \subseteq S_\Lambda$ of initial states and a set $F_\Lambda \subseteq S_\Lambda$ of final states, states in $I_\Lambda$, $F_\Lambda$ and $S_\Lambda$ are closed under $Z$-isomorphisms for $Z = atom(P_\Lambda)$ and $Z \subset H$, a program $P_\Lambda$ of vocabulary $H$, and a relation $\pi_\Lambda$ over $S_\Lambda$ is determined by $P_\Lambda$ such that $\{S_{i+1} | (S_i, S_{i+1}) \in \pi_\Lambda\} = \{S_i \oplus u | u \in \mathcal{U}(P_\Lambda, S_i)\}$.

The vocabulary of an Abstract Database Transformation Machine is constituted by a state vocabulary and a background vocabulary. If $\Lambda$ is an Abstract Database Transformation Machine, $Z$ is a finite set of elements from the base domains of $\Lambda$. Then taking sets $\mathcal{X}$ and $\mathcal{F}$ of dynamic nullary function names (i.e. variables) and a set of non-nullary function names in the vocabulary of $\Lambda$, respectively, then a set $\mathcal{T}$ of terms of $\Lambda$ is defined as the smallest set satisfying $t \in \mathcal{T}$ for all $t \in U$ with $atom(t) \subseteq Z$, $\mathcal{X} \subseteq \mathcal{T}$, and $f(t_1, ...t_n) \in \mathcal{T}$, where $f \in \mathcal{F}$ and $t_i \in \mathcal{T}$ $(i \in [1, n])$.

We still have to define the programs $P_\Lambda$ used in the definition above. For this we will need location operators. For this let $\mathcal{M}(D)$ be the set of all non-empty multisets over a domain $D$, then a *location operator* $\rho$ over $\mathcal{M}(D)$ consists of a unary function $\alpha : D \to D$, a commutative and associative binary operation $\odot$ over $D$, and a unary function $\beta : D \to D$, which define $\rho(m) = \beta(\alpha(b_1) \odot \cdots \odot \alpha(b_n))$ for $m = \langle b_1, ..., b_n \rangle \in \mathcal{M}(D)$.

Using a location function that assigns a location operator or $\perp$ to each location, an update multiset can be reduced to an update set. For convenience, we use $\Delta^{\Upsilon}(r)$ and $\ddot{\Delta}^{\Upsilon}(r)$ to denote an update set and an update multiset produced by executing a rule $r \in \mathcal{R}$ on structure $\Upsilon$, respectively. A term $t$ is a Boolean term iff $valueSet(t, \Upsilon) = \{true\}$ or $valueSet(t, \Upsilon) = \{false\}$.

For an Abstract Database Transformation Machine $\Lambda$, let $\Upsilon$ be a state of $\Lambda$, $\mathcal{T}$ and $\mathcal{X}$ be the sets of terms and variables of $\Lambda$, respectively, then $P_{\Lambda}$ is inductively defined by a set $\mathcal{R}$ of rules.

**Update Rule:** $f(t_1, ...t_n) := t_0$, where $f(t_1, ...t_n)$ and $t_0$ are terms in $\mathcal{T}$, with update set $\Delta^{\Upsilon}(f(t_1, ...t_n) := t_0) = \{(f(a_1, ...a_n), a_0)\}$, where $a_i \in valueSet$ $(t_i, \Upsilon)$ for $i \in [0, n]$.

**Conditional Rule:** **if** $t$ **then** $r$ **endif**, where $t$ is a Boolean term and $r \in \mathcal{R}$. The update set is $\Delta^{\Upsilon}($**if** $t$ **then** $r$ **endif**$) = \Delta^{\Upsilon}(r)$ if $valueSet(t, \Upsilon) = \{true\}$, otherwise $\Delta^{\Upsilon}($**if** $t$ **then** $r$ **endif**$) = \emptyset$.

**Forall Rule:** **forall** $x$ **with** $\varphi(x)$ **do** $r$ **enddo**, where $x \in \mathcal{X}$, $\varphi(x)$ is a Boolean term containing $x$ and $r \in \mathcal{R}$. The rule first produces the update multiset $\ddot{\Delta}^{\Upsilon}($**forall** $x$ **with** $\varphi(x)$ **do** $r$ **enddo**$) = \biguplus_{i \in [1,n]} \ddot{\Delta}^{\Upsilon}([a_i/x]r)$ where $\{x \mid valset(\varphi(x), \Upsilon) = \{true\}\} = \{a_1, ..., a_n\}$ and $[a_i/x]r$ means that variable $x$ is bounded to value $a_i$ within the rule $r$. Then this update multiset is reduced to the update set $\Delta^{\Upsilon}($**forall** $x$ **with** $\varphi(x)$ **do** $r$ **enddo**$)$.

**Choose Rule:** **choose** $x$ **with** $\varphi(x)$ **do** $r$ **enddo**, where $x \in \mathcal{X}$, $\varphi(x)$ is a Boolean term containing $x$ and $r \in \mathcal{R}$. $\Delta^{\Upsilon}($**choose** $x$ **with** $\varphi(x)$ **do** $r$ **enddo**$)$ $= \Delta^{\Upsilon}([a/x]r)$, where $a \in \{x | valueSet(\varphi(x), \Upsilon) = \{true\}\}$ and $[a/x]r$ means that variable $x$ is bounded to value $a$ within the rule $r$.

**Parallel Rule:** $r_1 \| r_2$ with $r_1, r_2 \in \mathcal{R}$ and the update set $\Delta^{\Upsilon}(r_1 \| r_2) = \Delta^{\Upsilon}(r_1) \cup$ $\Delta^{\Upsilon}(r_2)$.

**Sequence Rule:** $r_1; r_2$ with $r_1, r_2 \in \mathcal{R}$. In this case the update set is $\Delta^{\Upsilon}(r_1; r_2)$ $= \Delta^{\Upsilon}(r_1) \oslash \Delta^{\Upsilon}(r_2)$, which results from first applying $\Delta^{\Upsilon}(r_1)$ on structure $\Upsilon$ and then applying $\Delta^{\Upsilon}(r_2)$ on the resulting structure.

**Let Rule:** **let** $\theta(t) = \rho$ **in** $r$, where $\theta$ is a location function, $\rho$ a location operator, $t \in \mathcal{T}$ and $r \in \mathcal{R}$. This gives $\Delta^{\Upsilon}($**let** $\theta(t) = \rho$ **in** $r) = \Delta^{\Upsilon}([\rho/\theta(t)]r)$ with $l \in \{x | x \in valueSet(t, \Upsilon)\}$, and $[\rho/\theta(t)]r$ means that the location operator associated with $l$ is bounded to $\rho$ within the rule $r$.

**Queries as Database Transformations.** Let us now take a look at the fixed-point query language in [3] that has been used for defining views in WISs. In order to adapt this language to the formalism of ASSs we have to make some assumptions regarding the background. One of the domains used in the background should correspond to a set of abstract identifiers, so we can use a base type *ID* to refer to values in this domain. Identifiers in views will be interpreted as abstract surrogates for URIs.

Then we assume the existence of a tuple constructor $(\cdot)$ and a set constructor $\{\cdot\}$. Note that the minimum requirement for a background structure given in [2] already contains the existence of a pair-constructor that would be sufficient for building the tuple constructor on top of it.

Furthermore, let us assume that the database part in each state consists of relations and that identifiers are used for all tuples. These assumptions allow us to define *extended terms* (eterms), and each term will be typed. For each type $t$ we take a countable set of variables $V_t$. These sets are to be pairwise disjoint. Blurring the distinction between a value in a domain and a constant term that is always interpreted by that value, variables and constants of type $t$ become eterms of that type. In addition, each relation $R \in \mathcal{S}$ is an eterm of type $\{(\text{ident} : ID, \text{value} : t_R)\}$. For each variable $\imath$ of type $ID$ there is a term $\hat{\imath}$ of some type $t(\imath)$. If $\tau_1, \dots, \tau_k$ are eterms of type $t$, then $\{\tau_1, \dots, \tau_k\}$ is an eterm of the set type $\{t\}$. If $\tau_1, \dots, \tau_k$ are eterms of type $t_1, \dots, t_k$, respectively, then $(a_1 : \tau_1, \dots, a_k : \tau_k)$ is an eterm of the tuple type $(a_1 : t_1, \dots, a_k : t_k)$.

If $\tau_1, \tau_2$ are eterms of type $\{t\}$ and $t$, respectively, then $\tau_1(\tau_2)$ is a positive literal and $\neg\tau_1(\tau_2)$ is a negative literal. If $\tau_1, \tau_2$ are eterms of the same type $t$, then $\tau_1 = \tau_2$ is a positive literal and $\tau_1 \neq \tau_2$ is a negative literal. A *ground fact* is a positive literal without variables.

A *rule* is an expression of the form $L_0 \leftarrow L_1, \dots, L_k$ with a fact $L_0$ (called the *head* of the rule) and literals $L_1, \dots, L_k$ (called the *body* of the rule), such that each variable in $L_0$ not appearing in the rule's body is of type $ID$. A *logic program* is a sequence $P_1; \dots; P_\ell$, in which each $P_i$ is a set of rules.

Finally, a *query* $Q$ on $\mathcal{S}$ is defined by a type $t_Q$ and a logic program $\mathcal{P}_Q$ such that a variable *ans* of type $\{(\text{url} : ID, \text{value} : t_Q)\}$ is used in $\mathcal{P}_Q$. A *Boolean query* can be described as a query $Q$ with type $t_Q = \mathbb{1}$ assuming a trivial type $\mathbb{1}$ in the background with exactly one element in its domain.

A program $P_1; \dots; P_\ell$ is evaluated sequentially. Each set of rules is evaluated by computing an *inflationary fixed point*. That is, start with the set of ground facts given by a database as part of a state $S$. Whenever variables in the body of a rule can be bound in a way that all resulting ground literals are satisfied, then the head fact is used to add a new ground fact. Whenever variables in the head cannot be bound in a way that they match an existing ground fact, the variables of type $ID$ will be bound to new identifiers.

According to the result in [6] this language can of course be expressed by ADTMs, but due to its declarative nature we prefer to preserve it as a language for expressing the views in WISs. In order to link the views or any other query to ADTMs, which we will need for the transactional database transformations, we want to incorporate query expressions as some form of syntactic sugar into ADTMs.

This can be done by extending the definition of terms in ADTMs by expressions of the form $\langle\langle P \rangle\rangle(ans)$ with a program $P$ using the answer variable *ans*. Such a term will be interpreted by the value associated with *ans* after executing the logic program.

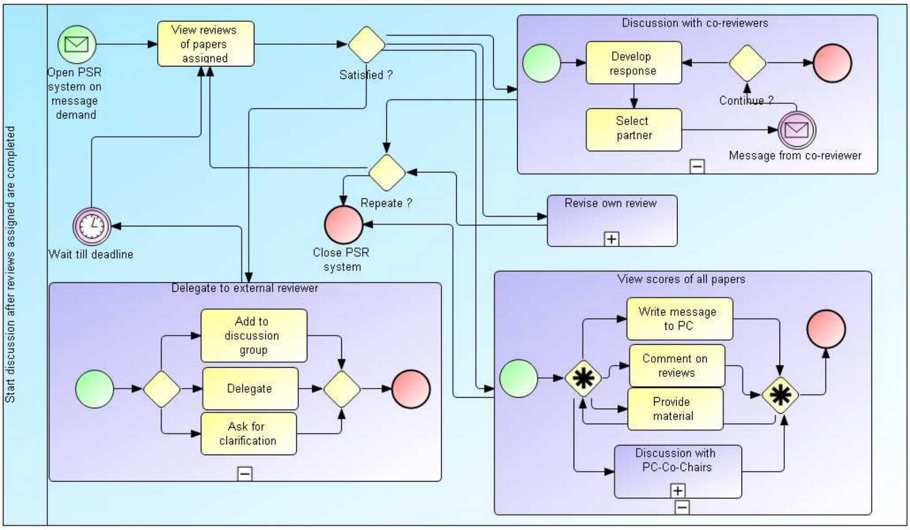



**Fig. 1.** The start of the discussion process for PC members

**Example.** Let us take a web-based conference management system (paper submission and reviewing PSR) as a familiar example. Let us further concentrate on the task of PC-members to discuss papers after they have been reviewed, and reviews of other reviewers for the same paper become visible. In this case an appropriate view would present the reviews and comments on a given paper, and the basic operations available to reviewers would be to enter a comment, and to revise the review. In order to do so, it may further be appropriate to obtain an anonymised overview of the ranking of all papers. If access rights permit, a discussant (usually a PC-co-chair, but the role of discussion convenor may also be created) may access all reviews of the participating PC-members to get an idea of their general attitude, i.e. whether they are usually hard or more generous towards authors.

The BPMN (Business Process Modelling Notation) diagram in Figure 1 displays the opportunities of the system for PC members. A PC member starts the subsystem and views first the reviews for the papers he or she reviewed. If the reviews are contradicting and the PC member wants to contribute to the discussion the discussion can either be directly started with all those PC members that have been reviewing at least one of the papers the PC member had or may revise the reviews accordingly or may delegate the discussion to the external reviewers the PC member asked or may first have a look onto all scores of the papers.

We left out the details of revision for own reviews. The diagram does not contain the message flow to other associated processes. The start process is only used for the start of the PC discussion. In a similar form we may model the discussion session of the PC. The diagram may also be enhanced by processes aiming in negotiation among reviewers of a paper and in mediating in the case of conflicts among the PC members.

Furthermore, the system may have a formalised treatment of subreviewers so that in case a review was actually done by a subreviewer, the PC-member has the choice to delegate the discussion to the subreviewer or to involve the subreviewer in the discussion or to only communicate with the subreviewer without letting him participate in the discussion. However, in order to keep things simple we will only add the subreviewer's name to a review. We also ignore the possibility of comments.

For such a system we could model a simple relational database schema. At its core – omitting all parts that are not relevant for the discussion task – we would have the following signature (using functions with arity in parentheses to represent relations): paper(7), member(9), assigned(2), review(17).

All these functions are dynamic and controlled, i.e. they can be updated by the system. The seven components of paper correspond to attributes such as paper_id, title, contact_email, password, abstract, submission_date and accept_code. The components of member correspond to attributes member_id, name, address, email, phone, rights, user_id, password, type. Components of assigned correspond to member_id and paper_id. The 17 attributes for review could be id, member_id, subreviewer, paper_id, submission_date, contribution, positive_aspects, negative_aspects, confidential_remarks, details, confidence, originality, significance, technical_quality, relevance, presentation, recommendation. Paper authors are handled separately.

Using these functions we may have paper(19,"Abstract State Services",kdschewe@acm.org,"dr0w33@p","...",10-04-08,und) = 1 in some state indicating that on 10 April 2008 a paper with title "Abstract State Services" and abstract "..." was submitted. The contact email-address is kdschewe@acm.org, the paper received the id 19, and the chosen password is "dr0w33@p". The acceptance of the paper is undecided.

Now take the query expressed by the following logic program:

$$
\begin{aligned}
\text{pap}(p,t,ab,R) &\leftarrow \text{paper}(i_p,(p,t,e,pw,ab,d,c));\\
\hat{R}(i,n,n',c,pos,neg,dc,co,o,s,q,r,pr,or) &\leftarrow \text{pap}(p,t,ab,R),\\
&\text{review}(i_r,(m,n,n',p,d,c,pos,neg,cf,dc,co,o,s,q,r,pr,or)),\\
&\text{member}(i_m,(m,n,a,e,ph,rg,u,pw,ty));\\
\text{ans}(i,(p,t,ab,\hat{R})) &\leftarrow \text{pap}(i,p,t,ab,R).
\end{aligned}
$$

This will produce a set of tuples, each of which represents a paper with its paper_id, title, abstract and the set of reviews associated with it. So each tuples represents the data presented to a PC-member when working on the paper discussion task.

An obvious service operation is revise_review, which would replace the review in the database – a simple assignment rule will be sufficient for this. In the view only the modified review would be replaced.

Another service operation is overview, which would create an additional view with an overview of the ranking of papers. We omit the details of the logic program needed for this view.

## 4 Conclusion

In this paper we used Abstract State Services (ASSs) as a formal abstraction of web services in a very general sense, capturing functional services, data warehouses and WISs. The fact that ASSs are grounded in the theory of Abstract State Machines (ASMs) shows the potential of ASMs to push Conceptual Modelling to new levels [9]. On the other hand, ASSs offer the opportunity to develop a theory for Meme media [10] thereby linking the theory to a very powerful practical instrumentarium for editing, extracting and composing services.

We exploited the language of ADTMs as a very powerful instrument for modelling ASSs, as it captures all database transformations encompassing queries and uopdates, and enriched it with integrated declarative query expressions. In doing so, we opened the pathway for WIS integration and composition along the lines discussed for ASSs in general, which will lead us to web interoperability. Nevertheless, the means by which ASSs can be composed still have not been explored in full detail, thus providing opportunities for further research. Furthermore, applying ASSs for web interoperability in practical case studies is another field that could be fruitfully explored.

## References


1. Benatallah, B., Casati, F., Toumani, F.: Representing, Analysing and Managing Web Service Protocols. Data and Knowledge Engineering 58(3), 327–357 (2006)
2. Ma, H., Schewe, K.D., Thalheim, B., Wang, Q.: Composing Personalised Services on top of Abstract State Services (submitted for publication)
3. Schewe, K.-D., Thalheim, B.: Conceptual Modelling of Web Information Systems. Data and Knowledge Engineering 54(2), 147–188 (2005)
4. Gurevich, J.: Sequential Abstract State Machines Capture Sequential Algorithms. ACM Transactions on Computational Logic 1(1), 77–111 (2000)
5. Börger, E., Stärk, R.: Abstract State Machines. Springer, Heidelberg (2003)
6. Wang, Q., Schewe, K.D.: Axiomatization of Database Transformations. In: Prinz, A., Börger, E. (eds.) ASM 2007, University of Agder, Norway (2007)
7. Altenhofen, M., Börger, E., Lemcke, J.: An Abstract Model for Process Mediation. In: Lau, K.-K., Banach, R. (eds.) ICFEM 2005. LNCS, vol. 3785, pp. 81–95. Springer, Heidelberg (2005)
8. Zhao, J., Ma, H.: ASM-based Design of Data Warehouses and On-line Analytical Processing Systems. Journal of Systems and Software 79(5), 613–629 (2006)
9. Börger, E.: Modeling Workflow Patterns from First Principles. In: Parent, C., Schewe, K.-D., Storey, V.C., Thalheim, B. (eds.) ER 2007. LNCS, vol. 4801, pp. 1–20. Springer, Heidelberg (2007)
10. Tanaka, Y.: Meme Media and Meme Market Architectures. IEEE Press, Wiley-Interscience, USA (2003)

# A Meta-model Approach to the Management of Hypertexts in Web Information Systems

Roberto De Virgilio and Riccardo Torlone

Dipartimento di Informatica e Automazione
Universitá Roma Tre, Rome, Italy
{devirgilio,torlone}@dia.uniroma3.it

**Abstract.** Modern Web Information Systems (WIS) are called to manage a huge amount of pieces of information, in a way that is often difficult to share, correlate and maintain. This is mainly due to the fact that the engineering of such systems is often based on proprietary and ad-hoc modeling languages and this makes hard the interoperability between different WISs. In this scenario, an important task of a modern Web application is the development of its navigation model, which provides an hypertextual view of the underlying data. It follows that the interoperability of Web applications relies often on the comparison, integration and exchange of information expressed in terms of hypertext structures. In this paper, we present a metamodel approach to the representation and development of navigational models of Web Information Systems supporting the definition of different models and the translations from one model to another. To this aim, we present a *General Hypertext Model* that covers a variety of Hypertext modeling methods and show how this metamodel can be profitably used in different application WIS scenarios.

## 1 Introduction

The Web has become a world-wide standard platform for information system development and the design of Web interfaces for data sources has imposed itself as a new and challenging database problem. This issue has justified a number of proposals, coming from the database area, for the the effective and efficient management "data-intensive" Web sites [2,3,11]. Other relevant works in the field have investigated the problem of extending traditional design methodologies and CASE tools for the development of these application [1,4,8,16]. From a software engineering point of view, the development of applications requires proper modelling methods in order to guarantee architectural soundness and system maintainability and it is widely recognized that the development of Web applications differs from that of traditional information systems [5,7]. Therefore, several proposals have been done for enhancing the capabilities of Web site design tools and introducing suitable design methodologies [6,8,13,15,20]. The various approaches have both similarities and differences. A common goal is supporting all the activities involved in the process of Web development, from conceptualization to maintenance. All of these solutions agree on the separation of the domain analysis from the specification of the navigation space structure, and the design of the user interface. Also, the existing methodologies tackle the modelling of a Web application using a model-based approach. Typically we observe a model that describes the

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 416–425, 2008.

content (domain) as a first step in the approach. On the basis of the content model ,the approaches help the designer to specify the navigation over that content. After constructing the Web interface, many of the approaches allow the designer to configure how this conceptual navigation structure is rendered with specific presentation details.

Unfortunately however, the various proposals are based on different models, methods and techniques. Furthermore, most Web modeling languages have been designed without providing a metamodel based, unifying view and are focused on notational aspects of the language. With no explicit metamodel representations however, it is difficult to define transformations and exchanges between heterogeneous models, since no common and sharable format is available. The Web Modeling Language (WebML) [8] is one example that does not rely on an explicit metamodel: it is implicitly defined within the accompanying tool WebRatio [1] in terms of a DTD (i.e., a grammar-like textual definition for specifying a structure for XML documents), models are represented internally in XML, and XSLT is used for structural transformations and code generation. In addition, due to the high growth of the commercial activities in the Internet, many systems are being implemented in very short periods of time, without support of appropriate tools. For this reason, Web applications are often of low quality and are difficult to maintain. In this scenario, we believe that a metamodel-based design approach strongly simplifies the interoperability between Web applications, by providing a framework for the definitions of different models and the translations from one model to another. In particular, transformation rules can be expressed in a more compact and readable way. For this reason, we propose a metamodel, called *General Hypertext Model*, that allows the definition of a hypertext structure on a content domain and covers a large variety of navigation model.

The requirements that this meta-model has to meet are the following: (i) *formalization*: it should describe a Web application and its components in a formal manner; (ii) *reproducibility*: it should describe a Web application and its components at an abstract level so that repeated executions/adoptions are possible for specific subject domains; (iii) *compatibility*: it should adhere to the available standards and specifications; (iv) *reusability*: it should allow the identification, isolation, decontextualization, exchange and re-use of the components of a Web application.

The rest of the paper is organized as follows. In Section 2 we discuss related works. In Section 3 we illustrate our approach and, in Section 4, we discuss some application scenarios where such an approach can play a central role. Finally, in Section 5 we draw some conclusions and sketch future work.

## 2 Related Work

In the Web engineering research field various modeling approaches have been proposed, aiming at supporting a model-driven development of Web applications. The modeling methods have their origins in different communities. Among them, we distinguish between the *data-oriented* and the *object-oriented* approaches. The former are mainly based on the Entity-Relationship (ER) model [9] and the Dexter Reference Model [14], such as RMM [15], Araneus [16], HDM [13], HDM lite [12] and WebML [8]. The latter rely on Object Modeling Technique (OMT) [18] and Unified Modeling Language (UML) [19], such as OOHDM [20], Baumeister et al. [6] and Conallen [10] approaches.

***Data-Oriented Approaches.*** The modeling formalism of HDM is one of the first methods to define the structure and interaction in hypermedia applications. It is based on concepts borrowed from the ER Model and from the Dexter Model, introducing new primitives as *units* (i.e. corresponding to *nodes*). An entity is a hierarchy of components, made of units. At the hypertext level, HDM models the application domain in the *hyperbase layer*, and manages the *access layer*, defining a set of collections that provide users with the patterns to access the hyperbase such as *index* and *guided tour*. HDM distinguishes three types of links: *structural* (connecting components), *perspective* (connecting units), and *application* links (connecting components and entities of the same or different type). RMM uses *slice units* and *slice relationships* to compose a navigational structure over the data (i.e. entity and relationship). Slices are presentation units which appear as pages of hypermedia applications. RMM specifies different access primitive such as *grouping*, *index*, *guided tour* and *indexed guided tour*. Araneus, like RMM, embraces content level, hypertext level, and presentation level but emphasizes the content and hypertext level, only. A unique characteristic of Araneus is that content and hypertext level are refined independently from each other. Similar to RMM, the content level of Araneus relies on the ER Model. Considering the hypertext level, the Hypertext Conceptual Design formulated by the Navigation Conceptual Model (NCM) is refined by the Hypertext Logical Design, using Araneus Data Model (ADM) as formalism, tailoring the design towards the Web. Likewise RMM, just structural aspects are considered for the content level as well as for the hypertext level. WebML [8] uses a hypertext model to describe how the data from the domain model is published in a hypertext format, by distinguishing *site views*, *areas*, *pages* and *content units*. It also uses access primitives as *index*, *scroller* and *filter*.

***Object-Oriented Approaches.*** OOHDM is based on a conceptual model represented by a class diagram. A Web application is a view on this conceptual model. Classes of these views represent *navigation classes*. Navigation objects can be grouped in different *navigation contexts*. The access to these navigation elements is made by navigation structures as *indexes* and guided *tours*. The abstract interface specifies interface objects that the user will perceive. OOHDM uses the *Abstract Data Views* model to define static aspects of the interface and a Statecharts based technique to model dynamic aspects. Other methods use similar constructs of OOHDM such as the navigational context for making node and link-based views over OO structural descriptions. The approach proposed by Baumeister et al. is based on OOHDM but instead of using a mix of different formalisms throughout the levels, UML is used as the basic modeling technique. Conallen defines a set of UML stereotypes for the Web modeling, but does not present a method for the design of Web applications. These stereotypes are appropriate to model layout and implementation aspects, but not to define the navigation structure of the Web application.

The first aim of this paper is to classify the main concepts of each method, considering the relationships between them and provide a metamodel for Web application definition. Figure 1 illustrates the different origins of these methods and the interdependencies between them [17].

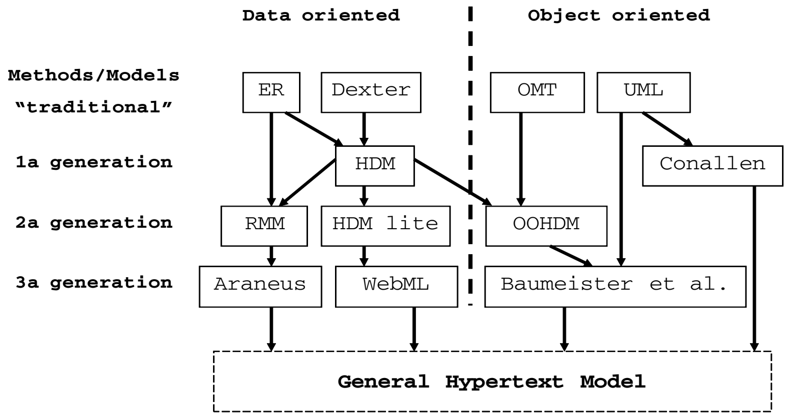



**Fig. 1.** Web Application Modeling

## 3 A Meta-model Approach

In this section we provide a common abstract representation of the various modeling approaches to the representation of the navigational structure of a Web application. In particular we identify the basic constructs of each model used to describe the organization of a Web site, and classify them into a set of meta-constructs that constitute our metamodel.

### 3.1 GHM: The General Hypertext Model

According to the classical organization of a Web resource (content, navigation, and presentation) [8], our model provides an hypertext definition describing a way to organize the elements of the content into a hypertext. Figure 2 presents a class diagram of our model. More precisely, it organizes a page in a set of *meta containers* related by *links*. A metacontainer is *atomic* if it represents an atomic portion of a linked page and includes a direct reference to the elements of a content from which it takes data. It can be *basic* if it shows information about a single object (e.g. an instance of an entity) or *multi* if it shows information about a set of objects. A metacontainer is a *linker* if it contains an anchor between containers. Otherwise a metacontainer is *complex* if it is articulated in other containers (atomic, complex or linker). So a page represents a complex container. Each metacontainer is identified by a *name* and presents different *properties* (e.g. the attributes of the content to include). We can surf the containers through several *navigational structures* such as *Indexes* to show a list of objects without presenting the detailed information of each one, *Guided Tours* to show commands for accessing the elements of an ordered set of objects, and *Entries* to show edit field for inputting values used for searching within a set of objects meeting a condition. Metacontainers do not exist in isolation. They have to be connected by links to form a hypertext structure. Links can be *contextual*, if they connect containers carrying some information from the source

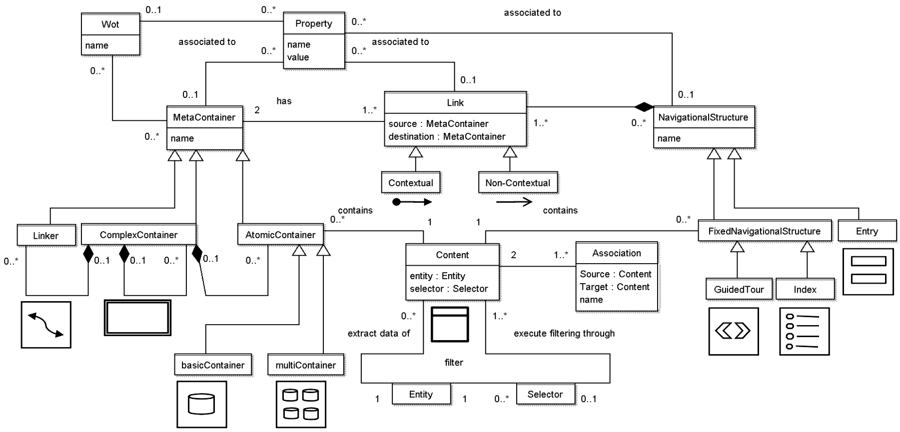



**Fig. 2.** UML diagram of our Web Site Model

container to the destination one (i.e. they have assigned properties in the model). Otherwise they are non-contextual if they connect containers in a totally free way. Finally, a *logical style sheet* describes the actual look and feel of metacontainers mentioned in a hypertext definition. It is based on a predefined set of *Web object types* (Wots). Possible Wots are text, image, video, form, and so on. Each Wot is associated with a set of presentation attributes (i.e. properties): they identify possible styles (e.g. font, color, spacing, position) that can be specified for it. Each attribute is associated with a domain of possible values for the attribute. Figure 2 illustrates the symbolic notation of each concept.

Moreover our metamodel provides a textual representation of an hypertext definition. It is a couple $\{MC, L\}$ where $MC$ is a set of metacontainers and $L$ a set of links between elements of $MC$. Each container has a textual representation as follows

**COMPLEX CONTAINER** $< Name >$
**PROPERTIES** $\{prop1, \ldots\}$
**CONTAINS** $\{mc_1, mc_2, \ldots\}$

**BASIC/MULTI CONTAINER** $< Name >$
**PROPERTIES** $\{prop1, \ldots\}$
**DATASOURCE** $< Name >$

**LINKER** $< Name >$
**PROPERTIES** $\{prop1, \ldots\}$
**CONTAINS** $\{l_1, l_2, \ldots\}$

Each link provides the following representation

**LINK** $< Name >$
**SOURCE** $< Name_{mc_1} >$
**DESTINATION** $< Name_{mc_2} >$
**PROPERTIES** $\{prop1, prop2, \ldots\}$

### 3.2 Representation of Web Models in GHM

Let us consider the main definition languages for database systems (HDM, RMM, WebML) and for object-oriented systems (OOHDM, Baumeister). We can describe principal constructs of each model in terms of GHM primitives, by a comparative concept based study. The resulting comparison is illustrated in Figure 3. It groups the design concepts of some of mentioned methods in the three typical design levels: conceptual, navigation and presentation level. These methods present a clear separation of the domain analysis. The Figure lists the most important concepts in each modeling level. At content level database systems use entity and relationship to design main data concepts and relations between them, while object oriented systems use class and oo-relationship. The main goal of this level is to select contents to organize in a hypertext. So GHM use content as a selection on entity (or class) and association as a relation (relationship or oo-relationship) between contents. At navigation level each method is characterized by link design, information nodes and access structure to navigate across nodes. Typically database systems distinguish between different types of link due to carrying parameters or not between nodes. Object oriented systems do not operate distinctions. GHM considers two types of links due to presence of parameters to carry or not. Therefore GHM uses non contextual links for object oriented systems. In the hypertext a node is a view on selected content. Database systems consider a node as a container that can organize information in different ways. For instance WebML distinguishes different units as data or multidata units. In object oriented systems a node is a class with attributes and methods. Finally both systems use different access primitives to surf across nodes as indexes, guided tour and so on. GHM uses the idea of metacontainer to represent different levels of detail. So an HDM node, a RMM slice or a WebML data unit are atomic containers while composition of nodes, slices, units or navigational classes are complex containers. At presentation level each method exploits HTML style elements. Therefore following them, GHM uses an idea of logical style sheet. It defines a set of multimedia types such as text, link, image, video and so on, and provides a set of presentation properties for each one. These properties determines the style rendering of elements in a hypertext definition using a mechanism similar to CSS one.

For instance, let us consider the example shown in Figure 4. It organizes a Web site referring to a management system of research paper reviews. In the Figure we show two different representations: the Baumeister (left side) and WebML (right side) models. Principally there is a main menu linking to Persons details (as Authors or as Reviewers) and Papers details across indexes. Then each Paper presents the Topic and the relative Reviews. Figure 5 illustrates the GHM representation of the same Web site. The hypertext definition shown is linked to the content level, expressed in terms of contents and associations, and to the presentation level, in terms of two Wots as Text and Link.

## 4 Application Scenarios

In this section we briefly illustrate some possible interesting scenarios in which Hypertext Management and the use of the General Hypertext Model can play a central role. In particular we describe the *Hypertext Definition Exchange*, the *Hypertext Definition Integration* and the *Hypertext Definition Adaptation*. The goal of this section is to provide interesting problems to which GHM is a valid instrument to support the solutions.

***Hypertext Exchange***. It recalls the well/know problem of data exchange. In that problem given a set of correspondences between a source and a target schema, a set of

| | HDM | RMM | WebML | OOHDM | Baumeister | GHM |
|---|---|---|---|---|---|---|
| Content Modeling | o entity<br>o relationship | | | o class<br>o oo-relationship | | o content<br>o association |
| Navigation Modeling | o **link**:<br>•Structural<br>•Application<br>•perspective | o **link**:<br>•unidirectional<br>•Bidirectional | o **link**:<br>•contextual<br>•non-contextual | o **link** | o **link** | o **link**:<br>•contextual<br>•non-contextual |
| | o node<br>o component | o slice | o site view, area, page<br>o unit (data, multi) | o navigation class<br>o navigation context | o navigational class<br>o navigational context | o metacontainer (complex, atomic, basic, multi, linker) |
| | o **navigation structure**<br>• collection<br>• index<br>• guided tour | o **navigation structure**<br>• grouping<br>• index<br>• guided tour<br>• indexed guided tour | o **navigation structure**<br>• index<br>• scroller<br>• filter<br>• direct | o **navigation structure**<br>• index<br>• guided tour | o **navigation structure**<br>• index<br>• guided tour<br>• menu | o **navigation structure**<br>• index<br>• guided tour<br>• entry |
| Presentation Modeling | o slot<br>o frame | | | o Abstract Data View<br>o inContext | | o Wot |

**Fig. 3.** A comparative concept-based study

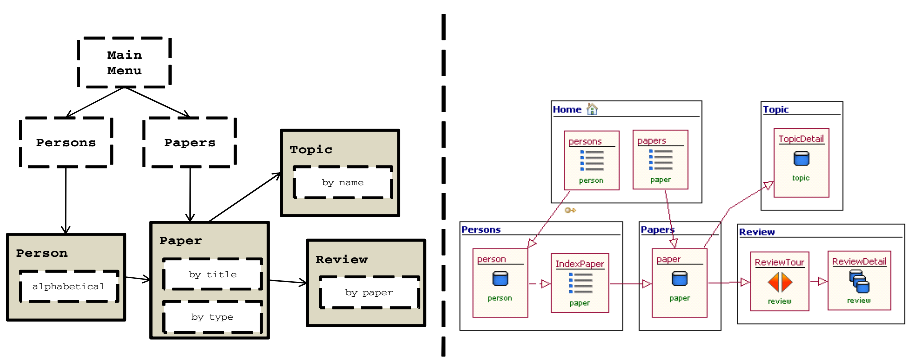



**Fig. 4.** Two Web modeling examples: Baumeister and WebML

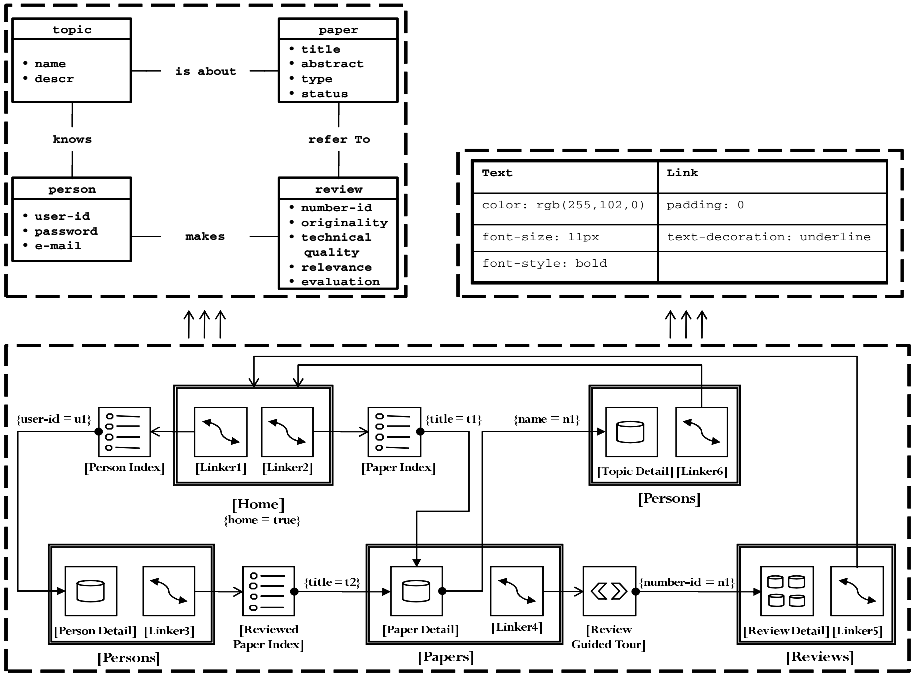



**Fig. 5.** The GHM representation of Review Web Site

queries have to be generated (automatically) to translate data from the source into a representation conforming to the target. In our scenario we consider a set of Web applications that present a similar Hypertext Structure. Then, given a set of correspondences between a source and a target Hypertext Structure we need a mechanism to translate any Hypertext schema conforming to the source into an Hypertext schema conforming to the target. Let's consider an online merchant selling books application. Many Web sites share a similar hypertext structure with respect to that domain. But the organization of some items (e.g. use of indexes or guided tours) can differ from an application to another. Moreover the applications can dynamically evolve over the time. So if we want to insert a new application into this domain we have to translate the structure of the new application into the common one. Therefore our metamodel supports this situation. We represent each Hypertext Schema (expressed in a particular model) into an Hypertext Definition and we define an Hypertext Structure as a set of couple $\langle \mathbf{mc}, \mathbf{pr} \rangle$ where **mc** is an occurring meta-construct and **pr** the set of properties of **mc**. So given a source and target Hypertext Structure, $HS_1$ and $HS_2$, an Hypertext Definition $H_1$ conforming to $HS_1$ and a set of correspondences between couples of $HS_1$ and $HS_2$, we generate an Hypertext Definition $H_2$ conforming to $HS_2$. The metamodeling approach permits to define a general translation mechanism (e.g by using production rules).

***Hypertext Integration***. Information integration is one of the oldest classes of applications. We face this problem in our scenario. Referring to the previous example of a Web application selling books, we would integrate two different Web sites that share an Hypertext Structure and then insert the integration into the common domain. In particular, given a source and target Hypertext Structure, $HS_1$ and $HS_2$, two Hypertext Definitions $H_1$ and $H_2$ conforming to $HS_1$ and a set of correspondences between couples of $HS_1$ and $HS_2$, we generate an Hypertext Definition $H_3$ conforming to $HS_2$ and containing all the information of $H_1$ and $H_2$. In this case we have to apply the Hypertext Definition Exchange to $H_1$ and $H_2$ generating two Hypertext definitions $H_1'$ and $H_2'$ conforming to $HS_2$ and then produce $H_3$ as the merge between $H_1'$ and $H_2'$.

***Hypertext Adaptation***. The proliferation of the rich, entertaining and interactive applications available on the Web justifies the need of a universal framework to anytime/anywhere access any data or information from any networked computer or device and by using any access network. Therefore this scenario challenges the ability of Web applications to adapt their behavior respect to the context of a client. In particular a Web application has to adapt its Hypertext Definition to the organization of a mobile device, to the irregular quality rate of the network and son on. Also in this case the metamodeling approach is a valid support for the task. It permits to define a general adaptation mechanism covering the existing modeling space.

## 5 Conclusion and Future Works

In this paper, we have illustrated a new approach to model Web applications. Various modeling languages in the Web engineering field are not based on metamodels and standards but rather constitute proprietary solutions mainly focused on notational aspects. An important task of these systems is the construction of a navigation model from the conceptual data model. The result is a hypertext definition, representing a view over a given database. In order to simplify this activity, we have proposed a metamodel that is an abstraction of existing formats for Hypertext applications. We have discussed several application scenarios in which our metamodel provides a valid support. There are some issues that need to be investigated to fully realize the potentiality of this approach. These include: (i) extending the set of basic primitives to capture a larger number of formats, (ii) the definition of a language to query and manipulate a hypertext definition, and (iii) the application of the approach to challenging real scenarios of use.

## References

1. Acerbis, R., Bongio, A., Butti, S., Ceri, S., Ciapessoni, F., Conserva, C., Fraternali, P., Toffetti, G.: WebRatio, an Innovative Technology for Web Application Development. In: Koch, N., Fraternali, P., Wirsing, M. (eds.) ICWE 2004. LNCS, vol. 3140, pp. 613–614. Springer, Heidelberg (2004)
2. Arocena, G.O., Mendelzon, A.O.: WebOQL: Restructuring Documents, Databases, and Webs. In: International Conference on Data Engineering, pp. 24–33. IEEE Computer Society, Florida (1998)
3. Atzeni, P., Mecca, G., Merialdo, P.: To Weave the Web. In: International Conference on Very Large Data Bases, pp. 206–215. Morgan Kaufmann, Athens (1997)

4. Atzeni, P., Mecca, G., Merialdo, P.: Design and Maintenance of Data-Intensive Web Sites. In: Schek, H.-J., Saltor, F., Ramos, I., Alonso, G. (eds.) EDBT 1998. LNCS, vol. 1377, pp. 436–450. Springer, Heidelberg (1998)
5. Atzeni, P., Torlone, R.: Management of Multiple Models in an Extensible Database design tool. In: Apers, P.M.G., Bouzeghoub, M., Gardarin, G. (eds.) EDBT 1996. LNCS, vol. 1057, pp. 79–95. Springer, Heidelberg (1996)
6. Baumeister, H., Koch, N., Mandel, L.: Towards a UML Extension for Hypermedia Design. In: France, R.B., Rumpe, B. (eds.) UML 1999. LNCS, vol. 1723, pp. 614–629. Springer, Heidelberg (1999)
7. Bernstein, P.A.: Applying Model Management to Classical Meta Data Problems. In: Proceedings of Biennial Conference on Innovative Data Systems Research, CA, USA (2003)
8. Ceri, S., Fraternali, P., Bongio, A., Brambilla, M., Comai, S., Matera, M.: Designing Data-Intensive Web Applications. Morgan-Kaufmann, San Francisco (2003)
9. Chen, P.: The Entity-Relationship Model - Toward a Unified View of Data. ACM Transactions on Database Systems 1(1), 9–36 (1976)
10. Conallen, J.: Modeling Web Application Architectures with UML. Communications ACM 42(10), 63–70 (1999)
11. Fernandez, M.F., Florescu, D., Kang, J., Levy, A.Y., Suciu, D.: Catching the Boat with Strudel: Experiences with a Web-Site Management System. In: International Conference on Management of Data, pp. 414–425. ACM Press, Washington (1998)
12. Fraternali, P., Paolini, P.: A Conceptual Model and a Tool Environment for Developing More Scalable and Dynamic Web Applications. In: Schek, H.-J., Saltor, F., Ramos, I., Alonso, G. (eds.) EDBT 1998. LNCS, vol. 1377, pp. 421–435. Springer, Heidelberg (1998)
13. Garzotto, F., Paolini, P., Schwabe, D.: HDM - A Model-Based Approach to Hypertext Application Design. ACM Transaction of Information Systems 11(1), 1–26 (1993)
14. Halasz, F.G., Schwartz, M.D.: The Dexter Hypertext Reference Model. Communications ACM 37(2), 30–39 (1994)
15. Isakowitz, T., Stohr, E.A., Balasubramanian, P.: RMM: A Methodology for Structured Hypermedia Design. Communications ACM 38(8), 34–44 (1995)
16. Mecca, G., Atzeni, P., Masci, A., Merialdo, P., Sindoni, G.: The Araneus Web-Base Management System. In: International Conference on Management of Data, pp. 544–546. ACM Press, Washington (1998)
17. Retschitzegger, W., Schwinger, W.: Towards Modeling of DataWeb Applications - A Requirements' Perspective. In: American Conference on Information Systems, Long Beach, California, USA (2000)
18. Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., Lorenson, W.: Object-Oriented Modeling and Design. Prentice-Hall, Englewood Cliffs (1991)
19. Rumbaugh, J., Jacobson, I., Booch, G.: The Unified Modeling Language Manual. Addison Wesley, Reading (1998)
20. Schwabe, D., Rossi, G.: The Object-Oriented Hypermedia Design Model. Commununications ACM 38(8), 45–46 (1995)

# An Approach to Creating Design Methods for the Implementation of Product Software: The Case of Web Information Systems

Lutzen Luinenburg[1], Slinger Jansen[2], Jurriaan Souer[1], Sjaak Brinkkemper[2], and Inge van de Weerd[2]

[1] GX, Wijchenseweg 111, 6538 SW Nijmegen, The Netherlands
{lutzen.luinenburg,jurriaan.souer}@gxwebmanager.com
[2] Department of Information and Computing Sciences, Utrecht University, The Netherlands
{s.jansen,s.brinkkemper,i.vandeweerd}@cs.uu.nl

**Abstract.** Development of a new software product is a complex process, in which requirements, release deadlines, and technical challenges fight for priority. These products are designed with the first release in mind which can sometimes cause software developers to have to make extensive design changes in subsequent releases. This paper presents an approach that creates design methods for product software design projects. The resulting design method is used to create a complete, maintainable, and durable product design. The approach has been applied in a practical case study of a series of web information system design projects. In this paper we describe the part of the case where key features were gathered and associated to method fragments, and how they were validated by an expert group.

## 1 Introduction

Today's software product vendors face several development challenges such as dealing with fast-changing business requirements, negotiating with a large number of stakeholders, and reducing the technical complexity of software products. While trying to find balance between these development challenges, software vendors also need to guarantee the success of their software product implementations in terms of efficiency and cost-effectiveness. When implementing a software product at a customer, a vendor has to once again deal with fast-changing requirements. In many implementation projects, a generic software product is adapted to and extended towards the needs of an organization. In addition, these requirements have a high impact on the architecture and design of an application. Once captured, it is hard to translate these business requirements into the pre-existing features of a software product. To make this translation transparent, flexible and less complex, design methods are needed for the implementation of software products. However, there are a myriad of software engineering domains and even more software products that are part of these domains. Additionally, it is a difficult task for software vendors to create a design method that meets the situational design context of a design project. When looking at the current body of knowledge in this field, no generic approaches or methods

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 426–436, 2008.

exist to support product software design with the aim to create durable and maintainable product software designs during implementation projects.

In this paper we propose an approach for design projects, using situational method engineering. Situational method engineering has been applied successfully in several cases and is an approach for finding the right method for an organization [1]. "Method engineering is the engineering discipline to design, construct and adapt methods, techniques and tools for the development of information systems" [2]. A special type of method engineering is situational method engineering which allows for creating methods that are tuned towards the unique development project situation [1, 20, 21]. Assembly-based situational method engineering is an approach in which method components are extracted and stored in a method base driven by situational method requirements [14, 20]. A method fragment is a description of an IS (Information Systems) engineering method, or any coherent part thereof [10].

The approach for design projects provides a guide in the selection and assembly of suitable method fragments into a product software design method. The approach helps a method engineer to assemble a design method, deriving development project situations, storing relevant methods, and selecting the right method fragments based on product-related situational factors. We call this approach the *Association and Assembly approach (AaAa).*

We continue this paper by presenting related work. In the third section we describe the AaA approach. Section 4 presents a WIS case and the derivation of key feature groups. Section 5 provides an overview of key feature groups established in the WIS case. The last section presents some comments and conclusions.

## 2 Creating the Association and Assembly Approach

Creating a design method that addresses every gathered functional requirement of a software product in a certain domain is complex. When looking at the domain of web engineering [6] many WIS design methods seem to be suitable for the design of data-intensive and process-intensive web applications. Examples of these methods are: OO-H [8], WebML [5], OOWS [13], Hera [22], and UWE [12]. However, these methods focus on web applications that are built from scratch without any pre-existing components and ignore domains in which pre-existing components are used to develop new applications. An example of a domain where pre-existing components are reused for new projects is the domain of Web Content Management Systems (WCMSs).

The Association and Assembly approach (AaAa) consists of the following steps:

1. Identify web engineering project situations
2. Identify feature groupings of a software product
3. Select candidate methods for the identified feature groupings
4. Model relevant method fragments in a method base
5. Associate feature groupings with candidate method concepts
6. Assemble situational web design method.

In order to address the fact that pre-existing components and functionality are available, we have adapted an existing assembly-based situational method engineering approach by adding two extra steps [20]. The first additional step (*step 2*) decomposes a software product into its main feature groups, in order to get more insight into its key features and product characteristics. The second additional step (*step 5*) creates an association table wherein selected method fragments and software product feature groups are formally compared.

In the first step, development project situations can be specified by means of categorizing unique development project characteristics. Basically, different development situations allow for the development of situational methods. For each development project situation, specific project needs can be defined, such as applying UML for modeling since all developers have knowledge of this modeling language.

Second, key feature groups of a software product domain are identified. These feature groups provide selection criteria for the assembly of a new method from a functional perspective.

Third, the domain of the software product is determined and in that domain relevant candidate methods are selected. This can be done by means of a literature study which provides an overview of the available methods. For scoping and relevancy purposes, it is important to narrow down the list of candidate methods into a comprehensive set of methods by means of useful selection criteria (e.g. acceptance in the community).

Fourth, method fragments of the selected candidate methods are extracted and stored in a method base. A method base structures the method fragments and thus captures knowledge about engineering methods [10]. For a uniform specification of all methods, a meta-modeling technique needs to be used which addresses the process as well the product perspective of a method [3].

Fifth, the selected candidate methods are compared by means of positioning the feature groups against the method fragments from a product perspective (e.g. deliverables in a meta-model) in an association table. This provides insight into which feature groups are addressed by the selected design methods. Next, it shows overlap between the methods in a way that a feature group is addressed by multiple method fragments. The association table also shows which feature groups are partly or not at all addressed by a method fragment.

In the final and sixth step, the results from the association table and the identified project needs are input for the assembly process in order to create a situational design method for the implementation of a software product. To ensure that all selected method fragments are assembled in a meaningful way, assembly guidelines should be taken into account. Brinkkemper et al. [3] have proposed logic rules for the assembly of a meaningful method addressing five quality requirements, such as completeness, consistency, efficiency, applicability and reliability. Having knowledge about feature groups allows a method engineer to analyze and select method fragments taking the feature groups and functionalities into account.

The final result of the AaA approach is a situational design method for the implementation of a software product taking the product's feature groups and development project situations into account. The resulting design method comprises of design activities and deliverables that allow for the translation of requirements into designed features that are ready for development in the follow-up phase. For each development situation, a specific route map can be chosen that influences the type and number of design deliverables produced during a project (*step 1*). Using this approach, a company can decide to use a more light-weight design approach for regular projects which enables them to produce design deliverables in an efficient way.

## 3 Derivation of Key Feature Groups in Practice: WIS Case

The AaA approach has been validated by means of a case study. This case has been performed within GX, a Web Content Management software company. GX develops and implements its proprietary Web Content Management System called GX WebManager [19]. As part of the Web Engineering Method (WEM), a situational WCMS design method has been constructed. WEM is a web engineering approach for the implementation of WCMSs and is consists of six traditional development phases: acquisition, orientation, definition, design, realization and implementation [17]. The AaA approach is evaluated through an expert validation and validated by practical projects according to Yin [23].

Souer et al. [17] positioned the term CMS-based web application which is a web application for the management and control of content. This type of web application can be seen as a subtype of a web information system and a WCMS. In figure 1, the implementation of AaA approach in a WCMS situation is depicted. In the first step, two development project situations have been defined, namely standard and complex development projects. Standard development projects were defined as development projects that did not need any customization from a component perspective. During the second step, fourteen key feature groups were identified.

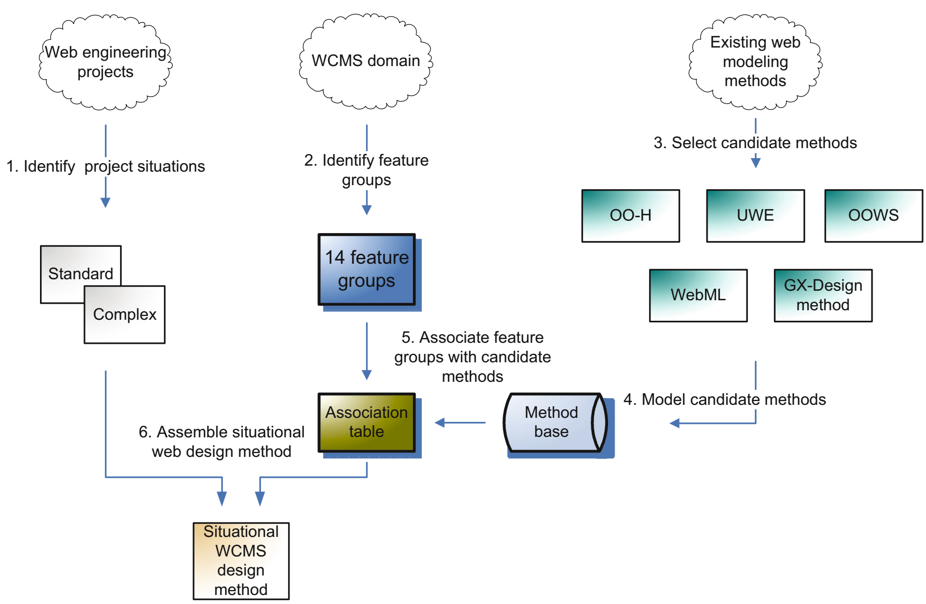



**Fig. 1.** The application of the Association and Assembly approach in a WIS case

These feature groups including the identification process are explained in the following sections. After the identification of feature groups, four mature WIS design methods and a propriety design method were selected. Three main selection criteria were applied: acceptance in the community, evolution of the approach and advanced tool support. In the fourth step, all relevant method fragments (e.g. requirements engineering, design and implementation fragments) were stored by means of using a meta-modeling technique called Process Deliverable Diagram (PDD) [20]. In total, 32 method fragments have been stored.

In step five, all candidate methods and key feature groups have been compared for selection purposes. During this association process it became clear that only three feature groups were well addressed by all methods: site management, personalization and e-forms. The domain and navigation models were suitable for these feature groups. Finally, a situational WCMS design method was assembled using the association table results and taking into account the development project needs. The situational WCMS design method consisted of four main activities: conceptual design, architecture design, presentation design and detail component design.

## 4 Key Feature Groups

Software product design addresses the requirements gathered during requirements engineering, therefore these product-related requirements have influence on the design of a software product. In this paper we call these design requirements feature groups. We define a feature group as "a class of functional design requirements." The purpose of such feature groups is to enable the association, selection and analysis of method fragments in order to assemble a situational method which is tailored mainly toward the design needs of a software product on a functional level. Please see the following section for more details on this process.

### 4.1 The Process of Establishing Key Feature Groups

No architectural reference models or frameworks are available in scientific literature concerning WCMSs that can be used for decomposing requirements into feature groups [17]. There is not much scientific literature present about WCMSs within the domain of web engineering. When analyzing the market of WCMSs and comparing it to the current state of literature, it can be argued that the business is ahead of the currently available scientific research. Therefore, the business of WCMSs is investigated in order to decompose a WCMS's functionalities into feature groups.

Despite the presence of many business selection tools, scientific well-founded reference models or standardized classifications of WCMSs are absent. Therefore, a marketing hype trend analysis has been performed in order to extract main feature groups from available market resources. Two of these WCMS feature groups are presented and described in the following section.

To gather key feature groups a literature review has been conducted wherein seven useful market resources have been identified. During this review step, a document analysis was also performed on the requirements and architectural documents of three complex development projects. The aim of this analysis was to find additional feature groups and to match already found features with feature groups implemented during development projects at GX. From these resources, all categories and features are derived which led to a total of 113.

Next, all features have been categorized based on occurrence. When a feature was mentioned more than once, it was listed. After this, the remaining features were analyzed and categorized by means of adding labels to the features. After labeling all the remaining features, all features were again counted and organized. During this categorization process, some features were left out since they were not relevant (e.g. speed). This step resulted in a list with 26 preliminary feature groups and their accompanying sub features. The last refinement step of all the feature groups was done by means of two expert validation interviews within GX. Based on the discussion and the comments, the list of feature groups has been refined and narrowed down to a final list of 14 feature groups.

## 5 A WCMS Feature Group Overview

In this section an overview of all feature groups identified during the WIS case is given. First, we elaborate on two of these fourteen feature groups, namely: e-forms / transaction management and personalization. The reason for elaborating on the personalization feature group is that is has been an important subject within literature in relation to e-commerce. Nowadays, personalization plays still an important role in web applications. E-forms are a means of user interaction and transactional services. The importance of this feature group has risen, since business processes run more and more via the web.

**E-forms / Transaction management** - E-forms are the means that allows a user to interact with a web application as well as its services and data. Moreover, e-forms enable companies to provide transactional services to customers in an easy way (e.g. application for insurance, banking, etc.). The design of e-forms requires a systematic approach to control the structure, appearance, semantics, user assistance information and validation logic [9]. A special language to design e-forms are XForms, which is an embeddable XML language, which means that it is designed to be incorporated into other XML languages. XForms provide a consistent and declarative structure for dynamic e-forms [4].

**Personalization** - Srivastava et al. [18] state that personalization of the web experience for a user is the holy grail of many web applications, pointing to individualized marketing for e-commerce. According the authors the following data categories can be collected for personalization: content, structure, usage and user profile. In conceptual WIS design modeling, personalization is about tailoring content, presentation, or navigation based on user preferences or user behavior [7, 11]. Besides, personalization on the web is often used for campaign management, such that context-aware or customized information is published for a particular user target group.

The other twelve identified feature groups are: (1) Authoring; (2) Authorization management; (3) Community technologies; (4) Connectivity management; (5) Content repository; (6) Deployment and replication; (7) Digital Asset Management; (8)

Layout and presentation management; (9); Multi-channel delivery and syndication; (10) Site Management; (11) Web usage Mining; and (12) Workflow.

### 5.1 Using Feature Groups for Association and Selection

After the identification of the key feature groups of a software product, the feature groups can be used for the association and selection of relevant method fragments which support the design of a software product (see step 5). The list of feature groups is made situational by means of mapping the feature groups onto the existing functionalities of the software product. The reason for doing this is that a design method of a software product must address the implementation of these groups in the product itself. This makes the situational method specific for a particular software product. The feature group e-form was translated in several functionalities of the WCMS like validation, step, handler, router and field.

Referring to figure 2, the navigation models of OOWS and OO-H are associated with the specialized e-form and personalization feature groups. From the association table, the most suitable method fragments were selected with regard to coverage of the feature groups. These method fragments were also mapped onto the project needs defined in the first step of the AaA approach. The situational method has been assembled by means of creating relationships between all method fragments. For instance, concepts of the domain model were connected to navigation model concepts in order establish a mapping between those concepts. During the assembly of the situational method, we have applied assembly guidelines (e.g. consistency) in order to raise the quality of design [3].

For the association and selection of method fragments, an association table is created which positions feature groups and their product specializations against method fragments on three granularity layers: method, model and concept [3]. By comparing WCMS key feature groups with method fragments, a functional overview of all methods included from a WCMS perspective is obtained. Next to that, the association table gives insight into the coverage of all feature groups by several methods and therefore

| | | OOWS | | | | | | | | | | OO-H | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Navigational model | | | | | | | | | | NAD | | | | | | |
| | | Navigational map | Context | Navigational link | AIU | Search mechanism | Navigational class | Navigational relationship | Process context | Activity Container | Main-AIU | navigational target | Navigational node | Service node | Collection node | Class node | Link | OCL expression |
| **E-Form** | Step | | x | | x | | x | | x | x | x | | | | | x | | |
| | Handler | | | | | | | | x | | | | | x | | | | |
| | Validation | | | | | | | | | | | | | x | | | | |
| | Router | | x | x | | | | | x | | | | | | | | | |
| | Field | | | | | | x | | | | | | | | | | | |
| **Personalization** | Personalization rule | x | x | | | x | | | | | | | | | | | x | x |
| | User profile | | | | | | | | | | | | | | | | | |
| | User access | x | | | | | | | | | | | | x | | x | | |

**Fig. 2.** The association table (excerpt)

supports the selection of relevant method fragments. At the same time, the association table provides a path of evolution for the existing WIS design methods as well for situational design methods within a particular product domain.

### 5.2 Validation

We have applied two types of validation. First, the situational WCMS design method has been validated by means of an expert validation. Second, the complex route map of the conceptual design activity has been evaluated in two projects.

During the expert validation the completeness and correctness was evaluated in order to improve the design. The expert group consisted of two software architects and two consultants. Based comments provided by the experts, the design of the method has been improved.

The situational WCMS design method has been applied in a practical case study of a series of web information system design projects. The projects have been performed at a large Dutch Telecommunications Provider (Tel) and at a Dutch Governmental Organization (Gov). For the Tel project, sixteen use cases regarding a personal online space have been designed from retrospective, and for the Gov project, two complex use cases regarding an intranet web application have been designed conceptually.

Two consultants and three software developers were involved during the design phase. After creating the conceptual models, the project members filled out a survey. In the survey, questions were asked per conceptual design deliverable, addressing the following aspects: readability, abstraction level, correctness, tool supportability and applicability. Only for the Gov project questions were asked about the design process relating to aspects such as structure, involvement and communication. Subsequently, informal interviews were held in order to discuss the situational WCMS design method addressing topics such as customer communication, usefulness, superfluity and redundancy of information.

From these two projects, we observed that the situational WCMS design method was positively valued by all project members. However, software developers and consultants had different perceptions on the usefulness of several conceptual design models. In contrast to the software developers, consultants were less positive about the domain model, since they considered the models as too complex to understand for customers. Though, both groups stated that the domain model was useful for designing components that require customization. Next, it was commented that user modeling should not be a separate design step, but integrated with domain modeling. The two consultants had different opinions about the supportability of navigation models. The Gov consultant on the one hand considered the navigation model as supportive in order to get a better understanding in the information structure of a web application. On the other hand, the Tel consultant questioned the added value of the navigation model concerning communication and conceptual design in general. Furthermore, the Business Process Diagram (BPD) was perceived as the most readable and useful model in comparison to all other models.

In figure 3, a BPD is given that represents an illness notification process for employees of the Gov project. A user can fill in an illness notification and when confirmed, and based on this request the WCMS tries to get user information from a DB,

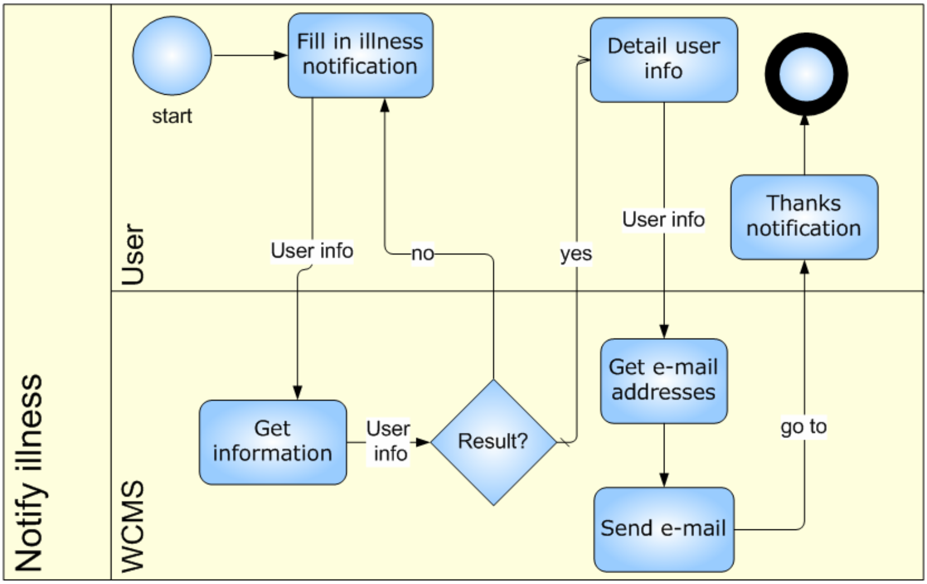



**Fig. 3.** A BPD that represents the 'notify illness process' of the Gov project

evaluates the results and based on that information, the user is able to provide additional user information. The process ends by retrieving relevant e-mail addresses for sending a notification to direct superiors of a particular employee.

## 6 Discussion

Before we can claim that the AaA approach is applicable to other domains, more case studies are required in different product domains. Currently, only one case has been used in order to validate the AaA approach.

Situationality was introduced into the generic design method for recurrent development projects in order to increase the method's applicability and productivity. We speculate that feature groups are related to product roadmaps, in that the product roadmap will include new feature groups, whereas feature groups might be able to assist in the creation of the product roadmap [15].

The AaA approach has been successfully applied to the assembly of a design method for recurring WIS design projects. The approach can provide a first design method to domains that are in need of more formal methods, tools or techniques.

## 7 Conclusions and Further Research

In this paper we presented an approach for creating a fitting software product design method for any type of software product implementation project. The approach provides a guide in the selection and assembly of suitable method fragments that create a product software design method. The approach helps a method engineer to assemble a method, derive development project situations, store relevant methods, and select the best-fitting methods based on product-related situational factors. We call this a feature group. Next, we showed how to use these feature groups for the association and selection of relevant method fragments. The approach was validated by applying it to one WIS design case.

Future research includes the investigation of applying metrics to the association table in order to support the selection of the right method fragments [16]. Finally, it is interesting to investigate the applicability of the AaA approach for the development of product line design methods.

## References


1. Aydin, M., Harmsen, F.: Making a Method Work for a Project Situation in the Context of CMM. In: Proceedings of the 4th International Conference on Product Focused Software Process Improvement, pp. 158–171 (2002)
2. Brinkkemper, S.: Method Engineering: Engineering of Information Systems Development Methods and Tools. Information and Software Technology 38(4), 275–280 (1996)
3. Brinkkemper, S., Saeki, M., Harmsen, F.: Meta-Modeling Based Assembly Techniques for Situational Method Engineering. Information Systems 24(3), 209–228 (1999)
4. Cardone, R., Soroker, D., Tiwari, A.: Using XForms to Simplify Web programming. In: Proceedings of the 14th international Conference on World Wide Web, pp. 215–224 (2005)
5. Ceri, S., Fraternali, P., Bongio, A., Brambilla, M., Comai, S., Matera, M.: Designing Data-Intensive Web Applications. Morgan Kaufman, San Francisco (2003)
6. Deshpande, Y., Murugesan, S., Ginige, A., Hansen, S., Schwabe, D., Gaedke, M., White, B.: Web Engineering. Journal of Web Engineering 1, 3–17 (2002)
7. Eirinaki, M., Vazirgiannis, M.: Web mining for Web Personalization. ACM Transactions on Internet Technologies 3(1), 1–27 (2003)
8. Gomez, J., Cachero, C.: OO-H Method: Extending UML to Model Web Interfaces. In: Information Modeling for Internet Applications, pp. 144–173. IGI Publishing, Hershey (2003)
9. Gouscos, D., Rouvas, S., Vassilakis, C., Georgiadis, P.: An Object-Oriented Approach for Designing Administrative E-forms and Transactional E-services. In: Bellahsene, Z., Patel, D., Rolland, C. (eds.) OOIS 2002. LNCS, vol. 2425, pp. 19–30. Springer, Heidelberg (2002)
10. Harmsen, F.: Situational Method Engineering. Ph.D thesis, University of Twente, Moret Ernst & Young Management Consultants, The Netherlands (1997)
11. Kappel, G., Proll, B., Retschitzegger, W., Schwinger, W.: Customisation for Ubiquitous Web Applications a Comparison of Approaches. International Journal on Web Engineering Technologies 1(1), 79–111 (2003)
12. Koch, N., Kraus, A.: The Expressive Power of UML-based Web Engineering. In: Schwabe, D., Pastor, O., Rossi, G., Olsina, L. (eds.) Second International Workshop on Web-oriented Software Technology, pp. 105–119 (2002)
13. Pastor, O., Fons, J., Pelechano, V., Abrahao, S.: Web Engineering: Theory and Practice of Metrics and Measurement for Web Development. In: Mendes, E., Mosley, N. (eds.) Conceptual Modeling of Web Applications: The OOWS Approach, pp. 277–301. Springer, New York (2005)

14. Ralyte, J., Deneckere, R., Rolland, C.: Towards a Generic Model for Situational Method Engineering. In: Eder, J., Missikoff, M. (eds.) CAiSE 2003. LNCS, vol. 2681, pp. 95–110. Springer, Heidelberg (2003)
15. Regnell, B., Brinkkemper, S.: Market-Driven Requirements Engineering for Software Products, Engineering and Managing Software Requirements. In: Aurum, A., Wohlin, C. (eds.) Engineering and Managing Software Requirements, pp. 287–308. Springer, Berlin (2005)
16. Rossi, M., Brinkkemper, S.: Complexity Metrics For Systems-Development Methods And Techniques. Information Systems 21(2), 209–227 (1996)
17. Souer, J., van de Weerd, I., Versendaal, J., Brinkkemper, S.: Situational Requirements Engineering for the Development of Content Management System-based Web Applications. International Journal on Web Engineering Technologies 3(4), 420–440 (2007)
18. Srivastava, J., Cooley, R., Deshpande, M., Tan, P.-N.: Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. ACM SIGKDD Explorations Newsletter 1(2), 12–23 (2000)
19. Van Berkum, M., Brinkkemper, S., Meyer, A.: A Combined Runtime Environment and Web-based Development Environment for Web Application Engineering. In: Persson, A., Stirna, J. (eds.) CAiSE 2004. LNCS, vol. 3084, pp. 307–321. Springer, Heidelberg (2004)
20. Van de Weerd, I., Brinkkemper, S., Souer, J., Versendaal, J.: A Situational Implementation Method for Web-based Content Management System-applications: Method Engineering and Validation in Practice. Software Process: Improvement and Practice 11(5), 521–538 (2006)
21. Van Slooten, K., Brinkkemper, S.: A Method Engineering Approach to Information Systems Development. In: Proceedings of the International Federation on Information Processing Working Group 8.1 Working Conference on Information System Development Process, pp. 167–186. North-Holland Publishing Co., Amsterdam (1993)
22. Vdovjak, R., Frasincar, F., Houben, G.J., Barna, P.: Engineering Semantic Web Information Systems in Hera. Journal of Web Engineering 2(1-2), 3–26 (2003)
23. Yin, R.K.: Case Study Research: Design and Methods. SAGE Publications, Thousand Oaks (2003)

# Semantic Verification of Web System Contents*

María Alpuente[1], Michele Baggi[2], Demis Ballis[3], and Moreno Falaschi[2]

[1] DSIC, Universidad Politécnica de Valencia
Camino de Vera s/n, Apdo. 22012, 46071 Valencia, Spain
alpuente@dsic.upv.es
[2] Dip. di Scienze Matematiche e Informatiche
Pian dei Mantellini 44, 53100 Siena, Italy
{baggi,moreno.falaschi}@unisi.it
[3] Dip. Matematica e Informatica
Via delle Scienze 206, 33100 Udine, Italy
demis@dimi.uniud.it

**Abstract.** In this paper, we present a rule-based specification language to define and automatically check semantic as well as syntactic constraints over the informative content of a Web system. The language is inspired by the GVERDI language and significantly extends it by integrating ontology reasoning into the specification rules and by adding new syntactic constructs. The resulting language increases the expressiveness of the original one and enables a more sophisticated treatment of the semantic information related to the contents of the Web system.

## 1 Introduction

Web systems are very often *collaborative* applications in which many users freely contribute to update their contents (e.g. wikis, blogs, social networks,...). In this scenario, the task of keeping data correct and complete is particularly arduous, because of the very poor control over the content update operations which may easily lead to data inconsistency problems.

In this paper, we propose a rule-based specification language which allows one to formalize and automatically check semantic as well as syntactic properties over the static contents of any Web system. The language is inspired by the GVerdi specification language [2,4] and extends it in the following ways.

**(i)** Web contents (typically, XML/XHTML data) are frequently coupled with ontologies with the aim of equipping data with semantic information. Our specification language provides ontology reasoning capabilities which allow us to query a (possibly) remote ontology reasoner to check semantic properties over the data of interest, and to retrieve semantic information which may be combined with the syntactic one for improving the analysis.

**(ii)** We extend the GVerdi specification language with new rule constructs for the definition of conjunctions and disjunctions of patterns which can be

* This work has been partially supported by the EU (FEDER) and the Spanish MEC under grant TIN2007-68093-C02-02, Integrated Action HA 2006-0007, UPV-VIDI grant 3249 PAID0607, and the Italian MUR under grant RBIN04M8S8, FIRB project, Internationalization 2004.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 437–446, 2008.

recognized inside XML documents. The new constructs increase the expressiveness of the original language, since they enable the specification of a larger set of semantic as well as syntactic constraints.

**(iii)** Along with the specification language, we formulate a novel verification methodology which automatically checks a specification against the considered Web contents and discovers incorrect as well as incomplete information.

**Related Work.** In recent years, several rule-based methodologies for validating the content of Web systems have been developed. In [6] constraint logic programming is applied to constrain the static content and the structure of a Web site, while [7] defines type systems and type checking techniques which are basically natural generalizations of DTDs and XML Schema definitions for describing and validating the structure of XML documents. Finally, the framework xlinkit [8] allows one to check the consistency of distributed, heterogeneous documents as well as to fix the (possibly) inconsistent information. All the mentioned approaches, albeit very useful in their specific domains, share the same limitation: the syntactic as well as semantic constraints they specify only rely on the data to be checked. Our approach tries to overcome such a limitation, on one hand, by introducing external sources of information (i.e. ontologies), which enrich the Web system content with additional semantic data; on the other hand, by enabling ontology reasoning for refining the verification process.

## 2 Description Logics and Ontologies

Description Logics (DLs) are logic formalisms for representing knowledge of application domains and reasoning about it. In particular, they are suitable means to model the semantics of XML data by defining ontologies which endow the raw XML data with semantic information.

Ontologies are complex data structures which basically consist of concepts (classes), individuals (instances of classes) and roles (relations). An ontology can be queried to retrieve semantic data or to verify semantic properties related to a given XML repository.

In the following, we present a description logic language for querying ontologies, which extends the well-known description logic formalized within the OWL-DL[10] framework by (i) admitting the use of variables as placeholders for atomic concepts, roles and individuals; (ii) letting function calls compute atomic concepts, roles and individuals. Due to lack of space, we present a DL language without the description of the DL constructs. For a thorough discussion about them, please see [3].

Let $\mathcal{A}$ be the set of all atomic concepts, $\mathcal{R}$ be the set of all atomic roles, $\mathcal{I}$ be the set of all individual names. Concepts are formalized via the following abstract syntax:

$$C \rightarrow A \mid X \mid f(t_1,\ldots,t_n) \mid \top \mid \bot \mid \neg C \mid C \sqcap C \mid C \sqcup C \mid \forall R.C \mid \exists R.C \mid \geq_n R \mid \leq_n R$$

where $C$ represents a (complex) concept, $A$ an atomic concept, $R$ a role, $n$ a natural number, $X$ a concept variable, $f(t_1,\ldots,t_n)$ a call to the function

$f: \mathcal{D} \mapsto \mathcal{A}$ that maps elements of a given domain $\mathcal{D}$ into the set of atomic concepts $\mathcal{A}$.

Roles can be described using the following abstract syntax:

$$R \rightarrow S \;|X \;|f(t_1, \ldots, t_n)$$

where $R$ represents a role, $X$ a role variable, $S$ an atomic role, $f(t_1, \ldots, t_n)$ a call to the function $f: \mathcal{D} \mapsto \mathcal{R}$ that maps elements of a given domain $\mathcal{D}$ into the set of atomic roles $\mathcal{R}$.

Finally, individuals are formalized via the following abstract syntax:

$$I \rightarrow a \mid X \mid f(t_1, \ldots, t_n)$$

where $a$ represents an individual name, $X$ an individual variable, $f(t_1, \ldots, t_n)$ a call to the function $f: \mathcal{D} \mapsto \mathcal{I}$ that maps elements of a given domain $\mathcal{D}$ into the set of individual names $\mathcal{I}$.

Concepts, roles and individuals, which are built by applying the grammar rules given above, can be used to query ontologies using the standard query constructs of Table 1. Therefore, given an ontology *ont* and a query construct *cst*, an *ontology query* on *ont* is an expression of the form: $ont.cst(a_1, \ldots, a_n)$, where $a_i$, $i = 1, \ldots, n$, are the arguments of the query construct *cst*. An ontology query whose execution returns a boolean value (respectively, a set of values) is called *boolean* (respectively, *non-boolean*) ontology query.

**Table 1.** Ontology query constructs

| **Query construct** | **Description** |
|---|---|
| *allConcepts()* | All concepts defined in the ontology |
| *allRoles()* | All roles defined in the ontology |
| *allIndividuals()* | All individuals defined in the ontology |
| *satisfiable(C)* | Is $C$ satisfiable? |
| *subsumes*($C_1$, $C_2$) | Does $C_2 \sqsubseteq C_1$? |
| *disjoint*($C_1$, $C_2$) | Does $C_1 \sqcap C_2 \equiv \varnothing$? |
| *children(C)* | All concepts which are children of $C$ |
| *equivalents(C)* | All concepts which are equivalent to $C$ |
| *instances(C)* | All individuals which belong to $C$ |
| *instanceOf(a,C)* | Does $a$ belong to $C$? |
| *roleFillers(a, R)* | All individuals $b$ such that *R(b,a)* holds |
| *related(R)* | All pairs *(a,b)* such that *R(a,b)* holds |

*Example 1.* Let `++` denote the string concatenation operator. Assume an ontology `univ` modeling an academic domain is given, the boolean ontology query

$$\texttt{univ.instanceOf}(\texttt{X++Y}, \texttt{professor} \sqcap \texttt{female})$$

would check whether the individual, which is computed by concatenating the values associated with variables `X` and `Y`, is a female professor (i.e. an instance of the concept `professor` $\sqcap$ `female`).

**Ontology Query Execution.** Typically, ontology queries are sent to an ontology reasoner and then executed against an ontology of interest. Standard ontology reasoners are able to execute only ground, flat ontology queries, that is, queries without variables and function calls. Note that our language allows one to generate ontology queries containing variables as well as function calls. Thus, in order to execute such queries using a standard reasoner, we need to made the query ground and to evaluate all the functions calls before sending the query to the reasoner. This process of query manipulation will be made explicit in the following sections.

## 3 The Web Specification Language

Our specification language allows us to formalize and verify properties over the content of a Web system.

**Web Content Denotation.** Throughout this paper, we assume that the data to be checked are stored into an XML repository. Moreover, since XML data are provided with a tree-like structure, we model XML repositories as finite sets of *ground* terms (that is, terms not containing variables) of a suitable term algebra. In the following, we will also consider *Web templates*, which are terms of a non-ground term algebra, which may contain variables. Web templates are used for specifying patterns to be recognized in XML repositories. See [2] for more details.

**Web Specifications.** The language provides constructs for specifying two kinds of rules: *correctness* rules and *completeness* rules. The former describe constraints for detecting erroneous information into a given XML repository, while the latter recognize incomplete/missing information. Both kinds of rules may be *conditional*, that is, they can be fired if and only if an associated condition holds.

A *condition* is a finite (possibly *empty*) sequence $c_1, \ldots, c_n$, where each $c_i$ may be (i) a membership test w.r.t. a regular language of the form $X \in \texttt{rexp}$[1], (ii) an equation $s = t$, where $s$ and $t$ are expressions which may contain nested function calls to be evaluated w.r.t. a given term rewriting system (TRS) $R$ formalizing the considered functions[2], and (iii) a boolean ontology query.

Given a substitution $\sigma$, which takes an expression and replaces its variables with ground terms and a condition $C \equiv c_1, \ldots, c_n$, we say that $C$ *holds* for $\sigma$ iff each $c_i\sigma$ is ground and

- if $c_i \equiv X \in \texttt{rexp}$, then $X\sigma \in \mathcal{L}(\texttt{rexp})$, where $\mathcal{L}(\texttt{rexp})$ is the regular language described by $\texttt{rexp}$;
- if $c_i \equiv (s = t)$, then the equality $(s = t)\sigma$ holds in the TRS $R$.
- if $c_i \equiv B$, with $B$ boolean ontology query, then the execution of $B\sigma$ returns *true*[3].

[1] Regular languages are denoted by the usual Unix-like regular expression syntax.
[2] In our framework, equation evaluation is handled by standard rewriting [9].
[3] We assume that all the function calls appearing in $B\sigma$ are evaluated before executing the query.

Now, we are ready to introduce correction as well as completeness rules.

**Definition 1 (Correctness Rule).** *A* correctness rule *is an expression of the form*

$$\bigwedge_{i=1}^{n} \mathtt{l}_i \rightharpoonup \mathtt{error} \mid \mathtt{C}$$

*where each* $\mathtt{l}_i$ *is a Web template,* error *is a reserved constant,* C *is a condition.*

Informally, the meaning of a correctness rule $\mathtt{l}_1 \wedge \ldots \wedge \mathtt{l}_n \rightharpoonup \mathtt{error} \mid \mathtt{C}$ is as follows. Whenever an instance $\mathtt{l}_i\sigma$ of $\mathtt{l}_i$ for each $i \in \{1, \ldots, n\}$ is recognized in some Web page p, and the rule condition C holds for $\sigma$, then Web page p is signaled as an incorrect page.

*Example 2.* Consider an XML repository containing academic information along with an ontology univ modeling such a domain. Suppose we want to verify the following property: *if an associate professor has more than three Ph.D. students, then he cannot teach more than one course.* Then a possible correctness rule formalizing such a property might be

```
use './Ontologies/UniversityDomain' as univ
course(cId(X),professor(name(Y)))
∧ course(cId(Z),professor(name(Y))) ⇀ error |
  univ.instanceOf(Y, AssocProf ⊓ (≥3hasStd ⊓ ∀hasStd.PhDStudent)),
  X =/= Z
```

To define completeness rules, we need the following auxiliary notion. Given an expression $e$, by $Var(e)$ we denote the set of all the variable appearing in $e$.

**Definition 2 (Completeness Rule).** *A* completeness rule *is an expression of the form*

$$\bigwedge_{i=1}^{n} \mathtt{l}_i \rightharpoonup \bigvee_{j=1}^{m} \mathtt{r}_j \mid \mathtt{C}\ \mathtt{containing}\ \mathtt{ct}\ \langle \mathtt{q} \rangle$$

*where each* $\mathtt{l}_i, \mathtt{r}_j$ *are Web templates,* C *is a condition,* containing ct *is an optional clause, where* ct *is a ground term,* $\mathtt{q} \in \{\mathtt{A}, \mathtt{E}\}$*, and* $\bigcup_{j=1}^{m} Var(\mathtt{r}_j) \cup Var(\mathtt{C}) \subseteq Var(\mathtt{l})$.

Completeness rules are called *universal* (resp., *existential*), whenever $\mathtt{q} \equiv \mathtt{A}$ (resp., $\mathtt{q} \equiv \mathtt{E}$).

Intuitively, given an XML repository $W$, the interpretation of a universal (resp., existential) rule $\mathtt{l}_1 \wedge \ldots \wedge \mathtt{l}_n \rightharpoonup \mathtt{r}_1 \vee \ldots \vee \mathtt{r}_m \mid \mathtt{containing}\ \mathtt{ct}\ \langle \mathtt{A} \rangle$ (resp., $\langle \mathtt{E} \rangle$) w.r.t. $W$ is as follows: if an instance $\mathtt{l}_i\sigma$ of $\mathtt{l}_i$ for each $i \in \{1, \ldots, n\}$ is recognized in some $p \in W$ and the condition C holds for $\sigma$, then an instance $\mathtt{r}_j\sigma$ of at least one $\mathtt{r}_j$, $j \in \{1, \ldots, m\}$ must be recognized in *all* (resp. *some*) XML documents of $W$ containing the ct term. Roughly speaking, ct provides the "scope" of the quantification and allows us to compute the part of the XML repository which is checked by the rule; if ct is not specified the rule is applied to the whole repository.

*Example 3.* Consider again an academic XML repository along with the usual ontology `univ`. We want to verify that *for each course, given by a full professor, at least two exam dates must be provided.* A completeness rule formalizing this property might be

```
use './Ontologies/UniversityDomain' as univ
course(cId(X)) ⇀ course(cId(X),examDate(),examDate()) |
                 univ.instanceOf(X,∃CourseGivenBy.FullProf) ⟨E⟩
```

Finally, we define a *Web specification* as a triple $(I_N, I_M, R)$, where $I_N$ is a set of correctness rules, $I_M$ is a set of completeness rules and $R$ is a TRS which defines the functions appearing in the conditions of the rules.

**Web Specifications with Meta-symbols.** Sometimes, it is particularly fruitful to consider rules containing Web templates which may subsume several meanings. To this purpose, completeness and correctness rules may include special *meta-symbols* into those Web templates which are associated with non-boolean ontology queries.

Web specification rules containing meta-symbols have to be pre-processed before being executed on a given XML repository.

The following definition is auxiliary. Let $e$ be a (syntactic) expression of our language, $m$ be a meta-symbol, $v$ be a symbol. By $e[m/v]$ we denote the expression $e'$ obtained from $e$ by replacing each occurrence of $m$ with $v$.

Basically, we expand each rule $r$ containing meta-symbols as follows:

- for each meta-symbol $m$ appearing in $r$, we execute the associated ontology query and we collect the results $\{v_1, \ldots, v_n\}$;
- if $m$ appears in the left-hand side of $r$, we replace $r$ with the rules $r_1[m/v_1]$, $\ldots$, $r_n[m/v_n]$.
- if $m$ appears in a disjunct $\rho$ of the right-hand side of $r$, we replace $\rho$ in $r$ with $\rho[m/v_1] \vee \ldots \vee \rho[m/v_n]$.

A full description of the expansion algorithm is available in [1].

For the sake of clarity, let us see an example.

*Example 4.* Consider again an academic XML repository along with the usual ontology `univ`. We want to specify that *email or post address have to be specified for each university professor.* Assume that the ontology `univ` contains (i) the concept `contactInfo` whose subconcepts are `email` and `address`; and (ii) the concept `Professor` whose subconcepts are `AssociateProf` and `FullProf`.
We might model the considered property using a universal completeness rule containing two meta-symbols (namely, `contact` and `prof`).

```
use './Ontology/UniversityDomain' as univ
metasymbol contact: univ.getChildren("contactInfo")
metasymbol prof: univ.getChildren("Professor")
prof(name(X)) ⇀ prof(name(X),contact())| containing member() <A>
```

By expanding the considered completeness rule, we generate the following set of rules without meta-symbols.

```
use './Ontology/UniversityDomain' as univ
AssociateProf(name(X)) ⇀ AssociateProf(name(X),email()) ∨
                         AssociateProf(name(X),address()) |
                                        containing member() <A>
FullProf(name(X)) ⇀ FullProf(name(X),email()) ∨
                    FullProf(name(X),address()) |
                                        containing member() <A>
```

## 4 Verification Methodology

In this section we present a methodology to automatically verify a given XML repository w.r.t. a Web specification. Without loss of generality, we only consider Web specifications without meta-symbols, since any Web specification with meta-symbols can be transformed into an equivalent one without meta-symbols as explained in Section 3.

We proceed as follows: first we describe the *partial rewriting* [2] mechanism which allows us to detect patterns inside XML documents and rewrite them. Then, we will employ this evaluation mechanism to check correctness and completeness of an XML repository.

**Simulation and Partial Rewriting.** Simulation allows us to recognize the structure and the labeling of a given Web template into a particular XML document. It can be formally defined as follows.

**Definition 3.** *The* simulation *relation* $\trianglelefteq$ *on terms is the least relation satisfying the rule* $\mathtt{f}(\mathtt{t_1},\ldots,\mathtt{t_m}) \trianglelefteq \mathtt{g}(\mathtt{s_1},\ldots\mathtt{s_n})$ *iff* $\mathtt{f} \equiv \mathtt{g}$ *and* $\mathtt{t_i} \trianglelefteq \mathtt{s}_{\pi(\mathtt{i})}$*, for* $\mathtt{i} = \mathtt{1},\ldots\mathtt{m}$*, and some injective function* $\pi : \{\mathtt{1},\ldots\mathtt{m}\} \rightarrow \{\mathtt{1},\ldots,\mathtt{n}\}$*.*

W.l.o.g., we disregard quantifiers from Web specification rules.

**Definition 4 (Partial rewriting).** *Let* s,t *be terms and* or*,* error *two fresh constructor symbols. We denote by* $\mathtt{s}_{|e}$ *the subterm of* s *rooted at position* $e$*. We say that* s partially rewrites *to* t *via rule* $r \equiv \bigwedge_{i=1}^{n} \mathtt{l}_i \rightharpoonup \mathtt{Rhs}|\mathtt{C}$ *and substitution* $\sigma$ *(in symbols* $\mathtt{s} \rightharpoonup_r \mathtt{t}$*) if and only if there exist positions* $u_1,\ldots,u_n$ *in* s *such that (i)* $\mathtt{l}_i\sigma \trianglelefteq \mathtt{s}_{|u_i}$ *for all* $i \in \{1,\ldots,n\}$*; (ii)* C *holds for* $\sigma$*; (iii) if* $\mathtt{Rhs} \equiv \bigvee_{j=1}^{m} \mathtt{r}_j$ *then* $\mathtt{t} \equiv \mathtt{or}(\mathtt{r}_1\sigma,\ldots,\mathtt{r}_m\sigma)$*; and (iv) if* $\mathtt{Rhs} \equiv \mathtt{error}$ *then* $\mathtt{t} = \mathtt{error}(\mathtt{s}, u_1,\ldots,u_n)$

**Detecting Correctness Errors.** Given a Web specification $(I_N, I_M, R)$, to detect erroneous or undesirable data included in an XML repository $W$, we have to execute all the correctness rule in $I_N$ against each XML document $p$ belonging to $W$. The procedure is described in the following definition.

**Definition 5.** *Let* $W$ *be an XML repository and* $(I_N, I_M, R)$ *be a Web specification. Given* $\mathtt{p} \in W$*, we say that* p *is* incorrect *w.r.t.* $(I_N, I_M, R)$*, if there exists a correctness rule* $r \equiv (\bigwedge_{i=1}^{n} \mathtt{l}_i \rightharpoonup \mathtt{error} \mid \mathtt{C}) \in I_N$ *such that*

*(i)* $\mathtt{p}$ *partial rewrites to* $\mathtt{error}(p, u_1, \ldots, u_n)$ *via* $r$ *and substitution* $\sigma$*;*
*(ii)* $\mathtt{C}$ *holds for* $\sigma$*. We say that* $(\bigwedge_{i=1}^{n} \mathtt{l}_i)\sigma$ *is an* incorrectness symptom *for* $\mathtt{p}$.

Note that the generated term $\mathtt{error}(p, u_1, \ldots, u_n)$ provides all the needed information to precisely locate the incorrectness symptom inside $p$.

**Detecting Completeness Errors.** The verification of an XML repository w.r.t. a set of completeness rules of a Web specification needs a more complex analysis. In the following, we introduce some semantic foundations we require to formalize the analysis.

A *completeness requirement* (or simply *requirement*) is a triple $\langle \mathtt{e}, \mathtt{q}, \mathtt{ct} \rangle$, where $\mathtt{e}$ and $\mathtt{ct}$ are ground terms and $\mathtt{q} \in \{\mathtt{A}, \mathtt{E}\}$. A requirement is called *universal* whenever $\mathtt{q} = \mathtt{A}$, while it is called *existential* whenever $\mathtt{q} = \mathtt{E}$. Sometimes the components $\mathtt{q}$, $\mathtt{ct}$ of a requirement can be left undefined, in this case we simply omit them and write $\langle \mathtt{e}, \_, \_ \rangle$. Such requirements are called *initial* requirements.

Let $\langle \mathtt{e}, \mathtt{q}, \mathtt{ct} \rangle$ be a requirement, $r \equiv \mathtt{Lhs} \rightharpoonup \bigvee_{\mathtt{j}=1}^{\mathtt{m}} \mathtt{r_j} \mid \mathtt{C}$ $\mathtt{containing}$ $\mathtt{ct_r}$ $\langle \mathtt{q_r} \rangle \in I_M$ be a rule such that $s \equiv \mathtt{e} \rightharpoonup_r \mathtt{or}(\mathtt{h_1}, \ldots, \mathtt{h_m})$. We define the tree $T_s$ associated with the partial rewriting step $s$ as $\mathtt{or}(\langle \mathtt{h_1}, \mathtt{q_r}, \mathtt{ct_r} \rangle, \ldots, \langle \mathtt{h_m}, \mathtt{q_r}, \mathtt{ct_r} \rangle)$.

**Definition 6 (Production Step).** *Let* $(I_N, I_M, R)$ *be a Web specification and* $\mathtt{re} \equiv \langle \mathtt{e}, \mathtt{q}, \mathtt{ct} \rangle$ *be a requirement. Let* $s_1, \ldots, s_k$ *be all partial rewriting steps which rewrite* $\mathtt{e}$ *using the rules in* $I_M$*. Let* $T_{s_1} \ldots, T_{s_k}$ *be the trees associated with the partial rewriting steps* $s_1, \ldots, s_k$*. The* production step *on* $\mathtt{re}$ *w.r.t.* $I_M$ *builds the tree* $\mathtt{re}(T_{s_1}, \ldots, T_{s_k})$.

Note that, if there is no rule $r \in I_M$ such that $\mathtt{e} \rightharpoonup_r \mathtt{t}$, we say that $\mathtt{re} \equiv \langle \mathtt{e}, \mathtt{q}, \mathtt{ct} \rangle$ is *irreducible*. Let us now use the production step to define the maximal derivation tree for a requirement.

**Definition 7 (Derivation Tree).** *Given a requirement* $\mathtt{re}$ *and a Web specification* $(I_N, I_M, R)$*, a* derivation tree *for* $\mathtt{re}$ *w.r.t. the set* $I_M$*, is defined as follows:*

- $\mathtt{re}$ *is a derivation tree for* $\mathtt{re}$ *w.r.t. the set* $I_M$*;*
- *if* $T$ *is a derivation tree for* $\mathtt{re}$ *w.r.t. the set* $I_M$ *and* $\mathtt{re}'$ *is a requirement labeling a leaf of* $T$*, then the tree* $T'$ *obtained from* $T$ *by replacing* $\mathtt{re}'$ *with the tree generated by applying a production step on* $\mathtt{re}'$ *w.r.t.* $I_M$*, is a derivation tree for* $\mathtt{re}$ *w.r.t. the set* $I_M$*.*

*A* maximal derivation tree $T_{\mathtt{re}}$ *for* $\mathtt{re}$ *w.r.t.* $I_M$ *is a derivation tree where all leaves are labeled with an irreducible requirement and the set of requirements labeling* $T_{\mathtt{re}}$ *nodes is finite.*

It follows that the maximal derivation tree for a requirement $\mathtt{re}$ w.r.t. $I_M$ contains all the requirements that can be derived from $\mathtt{re}$ w.r.t. $I_M$. Since the derivable requirements from $\mathtt{re}$ w.r.t. $I_M$ could be infinite, a maximal derivation tree for $\mathtt{re}$ might not exist. In [1], we propose some syntactical restrictions on Web specifications to ensure that the set of derivable requirements for a requirement

$\mathtt{re}$ w.r.t. a set of completeness rules is finite and hence a maximal derivation tree for $\mathtt{re}$ exists. Moreover, we provide a way to obtain from a maximal derivation tree, an equivalent finite structure.

Let $(I_N, I_M, R)$ be a Web specification, $W$ be an XML repository. An XML document $p \in W$ is associated with the initial requirement $\mathtt{re_p} = \langle \mathtt{p},\ \_,\ \_ \rangle$. Thus, we can compute the maximal derivation tree for $\mathtt{re_p}$ w.r.t. $I_M$, which contains *all* the requirements that can be derived from $p$ using the rules in $I_M$ by partial rewriting.

Now, to diagnose completeness errors in $W$, we can proceed as follows. For each $p \in W$, (i) we compute the maximal derivation tree $T_p$ for $\langle \mathtt{p},\ \_,\ \_ \rangle$ w.r.t. $I_M$, then (ii) we traverse $T_p$ and we check whether the requirements occurring in $T_p$ are not satisfied. The verification process terminates delivering the detected completeness errors.

**Definition 8 (Requirement Unsatisfiability).** *Let $W$ be an XML repository, $(I_N, I_M, R)$ be a Web specification and* $\mathtt{re} \equiv \langle \mathtt{e}, \mathtt{q}, \mathtt{ct} \rangle$ *be a requirement. Let* $\mathtt{TEST_{re}} = \{p \in W \mid \mathtt{ct} \trianglelefteq p\}$.

- *If* $\mathtt{re}$ *is a universal requirement, then* $\mathtt{re}$ *is* not satisfied *in $W$ if one of the following conditions hold:*
  1. $\mathtt{TEST_{re}} = \emptyset$*;*
  2. *there exists* $\mathtt{p} \in \mathtt{TEST_{re}}$ *s.t.* $\mathtt{e} \not\trianglelefteq \mathtt{p}$.
- *If* $\mathtt{re}$ *is an existential requirement, then* $\mathtt{re}$ *is* not satisfied *in $W$ if for each* $\mathtt{p} \in \mathtt{TEST_{re}}$*,* $\mathtt{e} \not\trianglelefteq \mathtt{p}$.

Dually, we can define universal/existential requirement satisfiability.

By using the previous definitions, we formalize the following completeness analysis.

**Definition 9 (Maximal Derivation Tree Analysis).** *Let $W$ be an XML repository, $p \in W$, $(I_N, I_M, R)$ be a Web specification, and $T_p$ be the maximal derivation tree of* $\langle \mathtt{p}, \_, \_ \rangle$ *w.r.t. $I_M$. The* completeness analysis *of a maximal derivation tree is inductively defined on the structure of $T_p$ by the following function:*

$$Verify(\mathtt{or}(T_1, \ldots, T_k)) = \begin{cases} \mathtt{error}(\mathtt{e_1}, \ldots, \mathtt{e_k}) & \textit{if } \forall i\ rt(T_i) \textit{ is not satisfied in } W \\ Verify(T_i)\ \forall i \textit{ s.t. } rt(T_i) \textit{ is satisfied in } W & \textit{otherwise} \end{cases}$$

*where* $rt(T_i) = \langle \mathtt{e_i}, \mathtt{q_i}, \mathtt{ct_i} \rangle$.

$$Verify(\mathtt{re}(T_1, \ldots, T_k)) = Verify(T_i)\ \forall i = 1, \ldots, k$$

The term $\mathtt{error}(e_1, \ldots, e_k)$ represents a completeness error which means that no disjunct $e_i$ for $i \in \{1, \ldots, k\}$ is recognized into the considered document. Roughly speaking, the execution of $Verify(T_p)$ finds all the completeness errors inside an XML document $p$ w.r.t. $I_M$. By applying this verification method to all XML documents in $W$, we can check the completeness of the whole $W$.

## 5 Conclusions

In this paper we presented a rule-based specification language, inspired by GVerdi [2,4], for formalizing and automatically checking semantic and syntactic properties over Web system contents. We increased the GVerdi language expressiveness by adding new constructs for defining conjunctive as well as disjunctive Web patterns. Moreover, we exploited ontology reasoning capabilities to retrieve semantic information which is useful to refine the verification process. In our framework, the integration between the language and the ontology reasoner is achieved by means of a description logic which extends the well-known OWL-DL[10] framework by admitting variables and function calls among the usual DL constructs. We are currently developing a prototypical system of our framework that implements ontology reasoning via the DIG Interface [5], which is an XML API for connecting applications to description logic reasoners.

## References


1. Alpuente, M., Baggi, M., Ballis, D., Falaschi, M.: Semantic Verification of Web Systems. Technical Report, University of Udine, Italy (2008)
2. Alpuente, M., Ballis, D., Falaschi, M.: Automated Verification of Web Sites Using Partial Rewriting. Software Tools for Technology Transfer 8, 565–585 (2006)
3. Calvanese, D., McGuinness, D., Nardi, D., Patel-Scheider, P., Baader, F.: The Description Logic Handbook. Cambridge University Press, Cambridge (2003)
4. Ballis, D., García Vivó, J.: A Rule-based System for Web Site Verification. In: 1st Int'l Workshop on Automated Specification and Verification of Web Systems. Electronic Notes in Theoretical Computer Science, vol. 157(2), pp. 11–17. Elsevier, Amsterdam (2005)
5. Bechhofer, S.: The DIG Description Logic Interface: DIG/1.1. Technical report, University of Manchester, United Kingdom (2003)
6. Florido, M., Coelho, J.: VeriFlog: Constraint Logic Programming Applied to Verification of Website Content. In: Shen, H.T., Li, J., Li, M., Ni, J., Wang, W. (eds.) APWeb Workshops 2006. LNCS, vol. 3842, pp. 148–156. Springer, Heidelberg (2006)
7. Pierce, B., Hosoya, H.: Regular Expressions Pattern Matching for XML. In: 28th Annual Symposium on Principles of Programming Languages. ACM SIGPLAN Notices, vol. 36(3), pp. 67–80. ACM, New York (2001)
8. Ellmer, E., Emmerich, W., Finkelstein, A., Nentwich, C.: Flexible Consistency Checking. ACM Transaction on Software Engineering 12(1), 28–63 (2003)
9. Klop, J.W.: Term Rewriting Systems. In: Handbook of Logic in Computer Science, vol. I, pp. 1–112. Oxford University Press, Oxford (1992)
10. World Wide Web Consortium (W3C). OWL Web Ontology Language Guide (2004), http://www.w3.org/

# Identifying Users Stereotypes with Semantic Web Mining

Sandro José Rigo[1,2] and José Palazzo Moreira de Oliveira[2]

[1] Universidade do Vale do Rio dos Sinos (UNISINOS) São Leopoldo RS Brasil
rigo@unisinos.br
[2] Instituto de Informática Universidade Federal do Rio Grande do Sul (UFRGS)
Porto Alegre RS Brazil
palazzo@inf.ufrgs.br

**Abstract.** This work describes the implementation of automatic user profile acquisition, using domain ontologies and Web usage mining. The main objective is the integration of usage data obtained from user sessions, with semantic description, obtained from domain ontology. In this way it is possible to identify more precisely the interests and needs of a typical user.

## 1 Introduction

During the last years, the Web continuous growth can be observed with the large document number and the flexibility in the access and interaction with Web applications. This scenario is a problem in daily tasks, as accessing and collecting desired data. One of the main reasons for this can be found in the one-to-many interaction model observed in traditional Websites.

Adaptive Hypermedia systems [1], [2] can reduce this problem by adapting content and structure of the Websites based on information about users profile, application domain and navigational behavior. Briefly, the actions to adapt the Website are based on the combination of the three aforementioned sets of information. The users profile information provides insights about adjustments that indicate feasible solutions. The application domain can be used to restrict or indicate some possible and coherent relations between the Website items. The navigational behavior provides current patterns, as observed from the actions of users accessing the Website. This information is processed together with adaptation rules or with pattern discovery techniques providing the system with elements to perform content or structure adaptation. The navigational patterns indicate specific and characteristic users behavior in one Website, allowing generalizations. When a pattern is identified as valid for an arbitrary number of users, then it may be considered in the generation of adaptations for new users. This is particularly important when the new user behavior is similar to this general pattern. The content and structure descriptions allow the detection of useful relations between the Website components. When a highly accessed Web page has relations with other pages, such as specialization or complementation, then

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 447–456, 2008.

these relations make possible the generation of relevant adaptations. Finally, the users interests and preferences can indicate the adequate characteristics for the content presentation and can also indicate the necessity for complementary content or for content restrictions, along with other possibilities.

However, in this approach usually the semantic information is not taken into account and this cause the results obtained not to present the possible quality. The lack of semantic information in a navigational pattern does not allow the identification of relations concerning structure, dependencies or complementary issues. The frequent access pattern describes the access sequence pages, but do not qualify and relates this accesses among themselves. This paper describes an Adaptive Hypermedia application that addresses solutions to the user profile acquisition problem. The adaptations are generated from classes of navigational behavior, identified with frequent accesses patterns improved with semantic information, described in domain ontology. The patterns identified allow the generation of adaptations that modify the structure of the Website, adding hyperlinks to possible users interests contents. These adaptations are generated based on usage information and semantic information relating the accessed pages, in order to describe tasks being performed by the users. Only the short term information is treated, such as the user session information. The process developed in this work is a general approach that can be applied to different Web applications.

The main contributions of this work are the improvements achieved with the integration of usage and semantic information concerning an application area. This integration process is performed without the user identification in the acquisition or in the adaptation steps. The objective of the system is the detection of tasks being performed by users in a specific session. With the semantic and usage information, the identification of a session probable objective can be used for evaluation of possible adaptations. Some problems related to users profiles also are avoided in this approach, such as the need for periodic updating or the limited set of information in the initial stages.

## 2 Related Works

Web Usage Mining applications focused on user profile acquisition can apply different algorithms, such as clustering [14], classification [15], association rules [3] and frequent sequences [16], [17], [18]. The main objective of most of these works is the efficiency in the algorithms implementation. Approaches considering semantic information are recent [19]. Some of these works generate the user profile with clustering and semantic annotation on the Web pages content [6],[4]. In others [5] , Web Content Mining is used to create taxonomies relations among Web pages. Web usage Mining is applied with the taxonomy terms to generate the users profiles. Domain ontology describing the Website structure can also be integrated with usage information. In the work of Mrabet [13], the users profiles are obtained with usage mining and domain ontologies describing the Website contents. The use of this information in combination with semantic information is reported in the work of Dai and Mobasher [6], where semantic information contained in the ontology of a Website is used together with the usage data for the generation of clusters and association rules. Thus the clusters and association rules generated allow integrate relevant details of each section of the site observed in the usage data.

There are methods for the identification of users based on the integration between Web usage and semantic information [7], [8]. In some works there is the use of clustering techniques that take into account the set of concepts identified in a group of pages that has been previously reported as a commonly followed path with associative rules or as a cluster [5], [4]. There can be found several techniques to acquire the user knowledge, abilities, goals and preferences. Most systems have identification interfaces and profile characterization facilities [9], [10], [11] while others seek to automatically obtain data that can be used in the creation or maintenance for the user model [12]. It has been shown, in this brief literature review, that there is a trend toward the use of semantic models from which are produced better adaptations to improve the Web user experience. The adaptations result is more interesting when complementary information is known, describing the application domain, the contents and the user profile. Most of the related works are content based or depend on user identification to drive the adaptations. The main differential of the present work is the exploration of semantically described Website structure and anonymous user navigation patterns.

## 3 General View of the Developed Architecture

The system implemented enables the integration of usage information and semantic information described in domain ontology. This integration allows identifying behavior classes, based on tasks carried out in the Website. These classes are associated with adaptation rules, which drive changes in the overall navigation structure of the Website. The first step in the process is the ontology description and the semantic content annotation. The system observes then the following periodic steps: the acquisition of Web usage data, the processing of these data for the generation of frequent access patterns, the integration of these patterns with semantic information, and, finally, the adaptation process. The application treats sequential access patterns, indicating the most frequent paths followed by the Website users.

As one of the objectives is the identification of tasks being performed in the Website, the domain ontology is defined manually by domain area experts, with respect to the Website structure and complementary relations. These relations are context dependent. Some experiments were performed in the educational area, and some examples for meaningful relations in this case are the precedence between elements, requisites or the content elements type.

The elements involved in the overall process of adaptation are illustrated in Figure 1. The figure illustrates that the requests generated by the Website users are stored and composes the “access log” which will be processed in the Usage Mining component. After the preprocessing of data and generation of

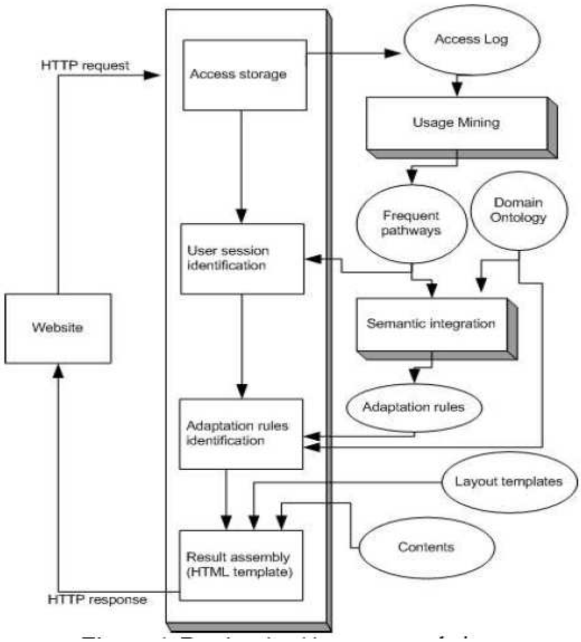



**Fig. 1.** Developed architecture general view

frequent access patterns, the system performs the integration of these pathways and semantic information. This result is evaluated and used to generate the adaptation rules that will be employed in the system. A cookie helps to identify when a user have any occurrence of access similar to the frequent access patterns already identified. This information will be used to generate the final result, composed by the content originally requested and by adaptation elements of the interface, in accordance with the detection of the access pattern.

The adaptation stage receives information from the domain ontology as well, in addition to the sequential patterns already obtained from the processing of the access log data. With these two sources of information, it is possible not only to identify the page access sequences, but also to identify the concepts present in each page and their relationship.

The information originated in the observed frequent sequential patterns is added to the original structure of the Website. Following the information already collected and processed, the system has a description of frequent sequential paths and specific relationships derived from the domain ontology. Based on the behavior observed from a user session the information is employed as complement to the original structure of the Website and is published in specific areas of the interface. Thus, the pages accessed by the user are correlated by sequential patterns. The identification that a certain number of pages accessed during a user session belong to a pattern can be assumed to indicate that the user in question is part of a group of users sharing a specific content or navigation interest, found in this set of pages.

The preprocessing of usage data and the generation of patterns is carried out periodically. During the users sessions, the system detects, from a recent access history that includes only accesses made in the current session, any coincidence of the observed behavior with the behavior patterns previously established and, if there exists such coincidence, carries out the associated adaptation. In these cases, as covered in the experiment, the adaptation was composed of alterations applied to the structure of the resulting page with the addition of new navigation possibilities derived from the patterns that have been established. Content adaptation could be carried out by the addition of material related to the content being accessed on the basis of some relationship given in the ontological description. The semantic annotation of the content permits the identification of specific situations, from the identification of the type of complementary resources.

## 4 Web Usage Mining and Semantic Information Integration

The usage information treated in this work is the frequent access patterns. The identified patterns are applied in the semantic integration. In the first step of this integration, each element in the access patterns is related with other elements in the pattern, based on the relations described in the domain ontology. This step allows the acquisition of semantic profile, containing the observed relations. The Figure 2 illustrates one example of this process. In the first part there is a typical access pattern, with Web pages identification. In the second part there are the relations observed in the domain ontology, when the Web pages identification is consulted. It can be observed relations as composedBy and partOf, indicating composition between pages and representing the Website structure. Can be observed relations indicating different element types and contents published, such as type, typeOfMaterial and typeOfContent.

In the second part of the Figure 2 can be observed that the Web pages identified as 49 and 50 are related with the composedBy relation, indicating that the Web page 49 is composed with the Web page 50. The Web pages 50 and 51 have several relations in common, as the typeOfMaterial with the value Classes and the typeOfContent with the value Programing Languages. These relations allow the detection of contexts existent in the frequent access patterns. As an example,

| Pageviews | Web pages identification |
|---|---|
| 3 | 49, 50, 51 |

| Pages | Relations observed |
|---|---|
| 49, 50 | (composedBy) [type-topic] |
| 50, 51 | [type-topic][partOf-ID_49][typeOfMaterial-TM_Classes]<br>[typeOfContent-Programing_Languages] |

**Fig. 2.** Access patterns and semantic patterns

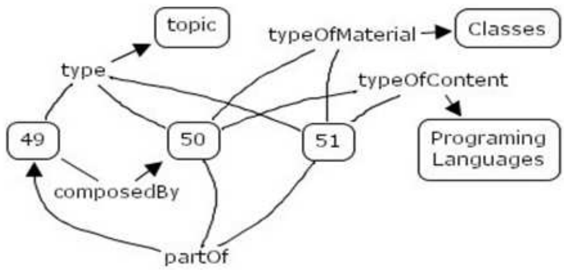



**Fig. 3.** Semantic context details

in the Figure 3 illustrate the same access pattern identified in the Figure 2, but describes the semantic information associated.

The semantic context can be applied to differentiate the access patterns and to generate resources to adaptation. The semantic information, described in the semantic context, can be applied to identify similar contexts. This can not be done with the access patterns only, because the lack of semantic information. In the case of a frequent access pattern that describes the browsing of a general page and its sub-pages, then this pattern can be identified as similar to the context of figure 3. The semantic context is generated with base on the domain ontology relations. These can be described regarding each specific application domain. The process applied in this work is independent of specific relations. Each access pattern L = {l1, l2, l3, ..., lm}, composed with m pages, is processed by SPARQL queries with each elements pair (li, li+1) and with each element and its relations with any other concept in the ontology.

To improve the adaptation possibilities the Website contents are semantically annotated. This is accomplished with the description of relations concerning structure and content description. In the work developed is applied a domain ontology. This decision implies in the description of the domain ontology for each new application. The validation of the proposal involved an application in the educational area. The domain ontology was described manually with the ontology editor Protg, and the ontology language OWL was used to their representation, helping posterior manipulation. In this case, the ontology represents the concepts, relations and content available on a Website with educational material.

There are relations between the concepts topic and course. The relations "partOf" and "composedOf" indicates the type of the composition. The relations "hasRequisite" or "isRequisiteOf" indicate dependency between the topics of one course. The relation "contain" allows to qualify each component in the Website. The relation between the Website and the ontology is established by the semantic annotation of the Website elements. Each page of the Website is described as an ontology instance, along with the necessary relations. These instances are used in the integration process along with the usage information.

| N | Patterns | Semantic contexts |
|---|---|---|
| 1 | 13, 12, 33 | (composedBy)(requisiteOf)[type-topic][typeOfMaterial-TM_ORGANIZATION]<br>(composedBy) [type-topic] [typeOfMaterial-TM_ORGANIZATION] |
| 2 | 12, 33, 44 | (composedBy) [type - topic] [typeOfMaterial - TM_ORGANIZATION]<br>(composedBy) [type - topic] [typeOfMaterial - TM_ORGANIZATION] |
| 3 | 49, 50, 51 | (composedBy) [type - topic]<br>[type - topic] [partOf - ID_49] [typeOfMaterial - TM_CLASS] [typeOfContent-linguagem_de_programacao] |
| 4 | 20, 46, 47, 48 | (composedBy) [type - topic] [typeOfMaterial - TM_CLASS]<br>[type - topic] [partOf - ID_20] [typeOfMaterial - TM_CLASS] [typeOfContent-algoritmo]<br>[type - topic] [partOf - ID_20] [typeOfMaterial - TM_CLASS] [typeOfContent-linguagem_de_programacao] |
| 5 | 49, 50, 51, 52 | (composedBy) [type - topic]<br>[type - topic] [partOf - ID_49] [typeOfMaterial - TM_CLASS] [typeOfCon-tent- linguagem_de_programacao]<br>[type - topic] [partOf - ID_49] [typeOfMaterial - TM_CLASS] [typeOfContent-linguagem_de_programacao] |
| 6 | 62, 64, 67, 72 | (requisiteOf) [partOf - ID_13] [typeOfMaterial - TM_ORGANIZATION]<br>(requisiteOf) [partOf - ID_13] [typeOfMaterial - TM_ORGANIZATION]<br>(requisiteOf) [partOf - ID_13] [typeOfMaterial - TM_ORGANIZATION] [requisiteOf - ID_78] |

**Fig. 4.** Obtained access patterns and semantic contexts

## 5 Experiments Analysis

The experiment in discussion was carried out over a period of ten months during which the material was available for users, with the necessary information for the adaptation being generated. Periodically the system generates frequent access patterns which are integrated with the semantic information described in the ontology. Each content item published is described in the domain ontology and this method allows the integration of usage and semantic information.

After the semantic integration process, the system can manipulate the semantic context obtained. In Figure 4 there are some examples of these contexts, ob-tained in the experiment performed. In the figure there is an association between the access pattern (displayed in the Patterns column) and the semantic context (Semantic Contexts column). Each context relates the component items of the access pattern. For instance, in line 1 there is an access pattern composed by the elements 13, 12 and 33. In the Semantic context column there is the information that the element 13 is composed by element 12 and is also requisite for this element. Both elements 13 and 12 are of type topic and their type of material is TM ORGANIZATION. The element 12 is composed by element 33 and they are also topic and TM ORGANIZATION elements.

The semantic contexts described in Figure 4 are analyzed in Figure 5. In this table there is a figure illustrating the structural relations of each pattern and there is also an interpretation of the context and the similarity with other contexts. This initial evaluation can indicate some advantages of the semantic context when compared with the access patterns. Some differences and similarities between the patterns will not be detected without the semantic contexts. This detection can be useful in the adaptation process, relation correctly the needs of a user session, with best suited information.

The patterns number 4, 5 and 6 present similarities when only the structural rela-tion are considered, as in the first column of the Figure 5. But with the analysis of the other relations, it can be seen that they have different function in the Website. The elements in the pattern 6 are associated with organization,

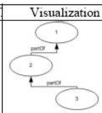
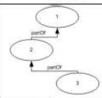
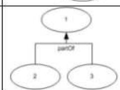
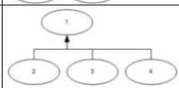
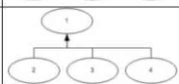
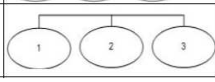
| | Visualization | Interpretation |
|---|---|---|
| 1 |  | The elements are organized with the "partOf" relation. |
| 2 |  | The elements are organized with the same relation "partOf" observed in the pattern 1. Their similarity is high. |
| 3 |  | There is a composition relation between the first and second element. The second and third share a composition relation with the first element, indicating contents composition. The similarity with patterns 1 and 2 is low. |
| 4 |  | There is a composition relation between the first and second element. The second, third and forth elements share a composition relation with the first element, indicating contents composition. The similarity with pattern 3 is higher than with patterns 1 and 2. |
| 5 |  | Similar to the pattern 4. High similarity with pattern 3. |
| 6 |  | This pattern indicates navigation in the first level of contents. Has low similarity with patterns 3, 4 and 5, associated with material type. |

**Fig. 5.** Visualization and interpretation of semantic contexts


**Fig. 6.** Example of structure adaptation

while the elements in the patterns 4 and 5 have the objective to describe classes material. There is another distortion that would not be detected concerning patterns 6 and 1. Despite their struc-tural differences, both have the same type of material, or the same function in the Website, which is detected with the very close similarity measure. In other cases, as in the elements 1 and 2, there is a very close visual and similarity measure result. The same occurs with patterns 3 and 4, even with different elements number. The adaptation step uses the semantic contexts and the information about the user session. This session information is obtained with a cookie that helps in the pathway identification. The adaptation possibilities are generated with the semantic context that shown higher similarity with the present session. Also the ontology relations concerning the accessed item can be used.

An example of the adaptation results can be observed in the Figure 6. Two areas were used for the generation of adaptation items, which are on the left, below the area from the main navigation menu and in the center, below the content area. In the first area can be seen the message "Suggested links" followed by a set of links related to the task identified by the adaptation rules. In the second can be seen the message "Topics requisites" followed by a link created with a query for information from the ontology. To generate data for the adjustments evaluation, the access to these items will be received by the system with an additional parameter.

## 6 Conclusion

This article presented an approach for the acquisition of user stereotypes based on Web Usage Mining and domain ontologies. In this case, the domain ontology describes important relations for the application. The relations in the ontology are combined with the usage patterns obtained by Web Usage Mining techniques. The integration of semantic with usage information was described and some related works were presented, in order to illustrate the possibilities of better identification of the interests and needs of a typical user with these resources. The main contribution of this work is the study of the possibilities for discovering navigational behavior related to tasks executed in a Website, adding semantic information and without the necessity of user identification. It is shown that adaptations using a domain ontology which associates Web pages with application concepts is useful and can lead to better results that those observed when only usage information is applied.

## References

1. Brusilovsky, P., Karagiannidis, C., Sampson, D.: Layered Evaluation of Adaptive Learning Systems. International Journal of Continuing Engineering Education and Lifelong Learning 14(4), 402–421 (2004)
2. De Bra, P., Arroyo, L., Chepegin, V.: The Next Big Thing: Adaptive Web-based Systems. Journal of Digital Information 5(1), 10–18 (2004)
3. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. In: 20th International Conference on Very Large Data Bases, pp. 487–499 (1994)
4. Mobascher, B., Dai, H.: Integrating Semantic Knowledge with Web Usage Mining for Personalization. In: Scime, A. (ed.) Web Mining: Applications and Techniques. Idea Group Publishing (2005)
5. Eirinaki, M., Vazirgiannis, M., Varlamis, I.: SEWeP: Using Site Semantics and a Taxonomy to Enhance the Web Personalization Process. In: 9th ACM Special Interest Group on Knowledge Discovery and Data Mining International Conference on Knowledge Discovery and Data Mining, pp. 99–108 (2003)
6. Mobasher, P., Dai, H.: A Road map to More Effective Web Personalization: Integrating Domain Knowledge with Web Usage Mining. In: International Conference on Internet Computing, pp. 58–64 (2003)

7. Jin, X., Zhou, Y., Mobasher, B.: Task-Oriented Web User Modeling for Recommendation. In: 10th International Conference on User Modeling, pp. 109–118 (2005)
8. Zhou, Y., Mobasher, B.: Web User Segmentation Based on a Mixture of Factor Analyzers. In: International Conference on E-Commerce and Web Technologies, pp. 11–20 (2006)
9. Petrelli, D.: User-Centred Design of Flexible Hypermedia for a Mobile Guide: Reflections on the HyperAudio Experience. User Modeling and User-Adapted Interaction 15(3), 303–338 (2005)
10. Wu, H.: A Reference Architecture for Adaptive Hypermedia Applications. Ph.D Thesis, Technische Universiteit Eindhoven, Eindhoven, ISBN 90-386-0572-2 (2002)
11. Staff, C.D.: The Hypercontext Framework for Adaptive Hypertext. In: 13th ACM conference on Hypertext and Hypermedia, pp. 11–20 (2002)
12. Dolog, P., Henze, N., Nejdl, W., Sintek, M.: Personalization in Distributed E-learning Environments. In: 13th International World Wide Web Conference, pp. 170–179 (2004)
13. Mrabet, Y., Khelif, K., Dieng-Kuntz, R.: Recognising Professional-Activity Groups and Web Usage Mining for Web Browsing Personalisation. In: IEEE/WIC/ACM International Conference on Web Intelligence, pp. 719–722 (2007)
14. Wang, K., Xu, C., Liu, B.: Clustering Transactions Using Large Items. In: 8th International Conference on Information and Knowledge Management, pp. 483–490 (1999)
15. Deshpande, M., Kuramochi, M., Wale, N., Karphys, G.: Frequent Sub-structure Based Approaches for Classifying Chemical Compounds. IEEE Transactions on Knowledge and Data Engineering 17(8), 1036–1050 (2005)
16. Han, J., Pei, J., Yin, Y., Mao, R.: Mining Frequent Patterns Without Candidate Generation: a Frequent Pattern Tree Approach. Data Mining Knowledge Discovery 8(1), 53–87 (2004)
17. Yan, X., Han, J., Xin, D.: Summarizing Itemset Patterns: A Profile-based Approach. In: 11th ACM Special Interest Group on Knowledge Discovery and Data Mining International Conference on Knowledge Discovery in Data Mining, pp. 314–323 (2005)
18. Xin, D., Han, J., Yan, X., Cheng, H.: Mining Compressed Frequent Pattern Sets. In: 31st International Conference on Very Large Databases, pp. 709–710 (2005)
19. Mei, Q., Xin, D., Cheng, H., Han, J.: Semantic Annotation of Frequent Patterns. ACM transactions on Knowledge Discovery from Data 1(3), 11–41 (2007)

# On Temporal Cardinality in the Context of the TOWL Language

Viorel Milea[1], Michael Mrissa[2], Kees van der Sluijs[3], and Uzay Kaymak[1]

[1] Faculty of Economics, Erasmus University Rotterdam
P.O. Box 1738, 3000 DR Rotterdam, The Netherlands
{milea, kaymak}@few.eur.nl
[2] University of Namur
Rue de Bruxelles 61, B-5000 Namur, Belgium
michael.mrissa@fundp.ac.be
[3] Eindhoven University of Technology
PO Box 513, NL-5600 MB Eindhoven, The Netherlands
k.a.m.sluijs@tue.nl

**Abstract.** The TOWL language is a temporal ontology language built on top of OWL-DL that enables descriptions involving time and temporal aspects such as change and state transitions. Extending OWL-DL into a temporal context does not only relate to providing the adequate expressiveness for such a goal, but also ensuring that static concepts preserve their meaning in a temporal environment. One such concept relates to cardinality. In this paper, we discuss temporal cardinality in the context of the TOWL language, and provide a possible approach towards representing temporal cardinality in this context.

## 1 Introduction

The role of Web Information Systems (WIS) on the Web is constantly increasing in importance. The Web 2.0 transformed Web pages from static pieces of text into dynamic applications with desktop-like functionality on a Web scale. It has thus become possible to exploit Web features, such as being able to integrate functionality of different applications into a new Web application. The Semantic Web adds a new dimension, as now not only functional blocks can be exchanged, but also knowledge, allowing information to be processed in entirely new ways. An important feature of the Semantic Web consists of its explicit semantics: metadata about the information is explicitly modeled in terms of classes and relationships between them. This provides rich descriptions that can be reasoned upon.

By using the state-of-the-art Semantic Web language OWL [1] we can, for example, model that an instance of the class *Person* is married with another instance of the class *Person*. If we want to be a bit more explicit we might consider modeling that a person is only married with one other person, i.e., that a marriage is symmetrical ($x$ is married to $y$, implies $y$ is married to $x$) and not all persons are married. In OWL this can be represented as follows.

I.-Y. Song et al. (Eds.): ER Workshops 2008, LNCS 5232, pp. 457–466, 2008.

```
ObjectProperty(:marriedTo Symmetric
    domain(:Person)
    range(:Person))
Class(:Person partial
    restriction(:marriedTo maxCardinality(1))
    restriction(:marriedTo minCardinality(0)))
```

However, this is a static model. If we want to be able to account for re-marriage, e.g. due to divorce or to the death of one of the partners, we might want to slightly change our restrictions and state that a person is only married to one other person *at any point in time*. Expressing such a restriction requires the existence of a construct for the representation of temporal cardinality. However, such a construct is not available in OWL, nor in its temporal extension, TOWL.

Different application domains motivate the need for a temporal ontology language. In this paper, we focus on two specific application domains: first, the trading domain (StockBroker), which is a highly dynamic application domain, and second, the cultural heritage domain (CHI). In StockBroker we utilize the Semantic Web to meet the increased technological demands emerging in the world of trading. The information that one seeks to represent in such a context relates mostly to news. With the increased popularity of the Web as a broadcasting medium, the latter has also become the main source of information and signals for financial traders. Different large players among multimedia news agencies already provide professional products that come to meet the increased need for tools supporting automated trading. Reuters, for example, provides a range of such products, such as the NewsScope Archive[1] – an annotated archive of news messages aimed at "customers seeking to develop news-based programmatic trading strategies with a comprehensive, machine-readable archive of Reuters global news." NewsScope Real-Time[2] is similar to the 'archive' version of this tool, with the main difference that the annotated feeds are provided in real-time, thus enabling automatic reactions to market-moving events. The StockBroker application is aimed at employing market knowledge from several Web sources for assisting in making better trading decisions.

CHI is a real-life application for Regional Historic Centre Eindhoven (RHCe), an institute that governs cultural heritage related to the region around the city of Eindhoven. The purpose of the CHI application is to open up a large dataset of multimedia documents with the help of additional metadata (in [2] we describe how to acquire this metadata by relating tags to concepts in an ontology). Time is a major dimension in this application, as most objects refer to a time in the past and since time is an essential aspect of the disclosure of the collections. A major issue with respect to time is to describe how we model change and evolution. For example, we represent locations of objects in our datamodel, say the location of the "city hall". If we have a picture that shows the city hall, we annotate this object with the "city hall" concept. The "city hall" concept has an address and the city hall coordinates so that we can show its location on a

[1] http://about.reuters.com/productinfo/newsscopearchive/
[2] http://about.reuters.com/productinfo/newsscoperealtime/

map. However, the location of "city hall" changes over time, e.g. because the old building is replaced or municipalities merge.

Possible problems we want to detect and bring to the attention of the RHCe domain experts are, for example, "This picture shows building X and has been annotated with date Y, but we also have in our knowledge base that building X was built on date Z and Z>Y, so this might be an inconsistency".

The most expressive fragment of the Web Ontology Language (OWL) that maintains desirable computational properties for reasoning is OWL-DL [1]. However, this language only provides limited support for the representation of time. The TOWL language [3], built as an extension of OWL-DL, comes to address this shortcoming by providing support for the representation of time and time-related aspects such as change and state transitions. In this paper, we discuss how the static OWL-DL concept of cardinality changes in a temporal context. More precisely, we discuss the concept of temporal cardinality in the context of the TOWL language, and provide a possible representation of the semantics of temporal cardinality when timeslices are employed for the representation of change, as provided by TOWL.

This paper is structured as follows. In Section 2. we provide an overview of related work. An overview of the TOWL language is given in Section 3. A discussion on the issue of temporal cardinality in the context of the TOWL language is given in Section 4. We discuss our results and give some insights for future work in Section 5.

## 2 Related Work

In this section, we overview existing works that tackle time-related issues for semantic representation. The representation of temporal semantics has for long been subject of investigation from the research community. It relates to several domains such as artificial intelligence, temporal databases or schema evolution [4]. The rise of the Semantic Web has lead to new attempts to integrate the temporal dimension into semantic languages. Some approaches choose to extend RDF triples (to quadruples) with time based information (e.g. [5]). The setbacks of extending RDF with time however are inflexibility and incompatibility: inflexibility in the sense that for every property a time interval has to be repeated while this might not be necessary by just using additional modeling primitives, incompatibility in the sense that no Semantic Web frameworks support quadruples which is why we would prefer to have a solution that chooses a modeling approach.

Many approaches in literature are built on description logic-based languages such as OWL [1] that have been widely adopted by the community, by extending these languages with temporal semantics in diverse ways [6,7,8,9]. These approaches rely on the notions of instant and time interval to describe temporal events.

In [6], an ontology of time is proposed for the semantic Web. This ontology is now accepted as a W3C working draft. As an extension to this work, the notion of temporal aggregates is introduced in [7]. Temporal aggregates are useful for describing recurrent events such as “every Tuesday”. The authors rely on set theory in order to describe temporal aggregates as ordered sets of time intervals.

In [8], the notion of complex temporal events is also introduced as a set of instants or time intervals, but the authors describe the temporal aspect of data with an object-oriented conceptual data model called MADS that is integrated into OWL. Time is represented with the help of MADS-OWL classes that link MADS constructs to OWL descriptions. The practical approach presented by the authors relates to spatio-temporal schema integration. The validation of schemas from a temporal perspective is performed by introducing additional properties and constraints over MADS-OWL classes, thus specifying and restricting the way information can be described. For instance, two classes with the same spatial properties are labeled with a “s_equal” property that expresses spatial equivalence.

In [9] an approach is presented for the description of fluents in OWL. In this context, fluents are nothing more than properties that hold between timeslices (temporal parts of an individual), and indicate what is changing. This approach builds on a perdurantist (or 4D) view where temporal objects are spatially represented over three dimensions, with time as the fourth dimension. This view is opposed to the previously described endurantist (or 3D) view that makes the distinction between objects and events. In a 3D view, objects remain always present and the temporal aspect is represented by different events occurring at certain times.

In the 4D view, temporal objects are described as *spacetime worms* made of slices, where each slice represents the state of the object at a time $t$. Spacetime worms have properties (fluents) that hold within certain time intervals (i.e. they span over several timeslices). This 4D view simplifies temporal representation as the time dimension is equally important to the other dimensions of the objects. Also, it has the benefit not to affect the use of several OWL constructs such as the inverse or transitivity operators. However, when it comes to cardinality representation, the fluent-based approach appears to be very verbose and suffers from several limitations. For instance, cardinality restrictions over the number of timeslices at a certain time cannot be represented. We tackle this problem and propose appropriate OWL constructs for cardinality description in the remainder of this paper.

## 3 The TOWL Language

The TOWL language [3] is an extension of OWL-DL that enables the representation of time and temporal aspects such as change and state transitions. It comes to meet some shortcomings of previous approaches, such as [6,9] that only address this issue to a limited extent and do not seek to enable automated reasoning in a temporal context. The language is designed by means of layers built on top of OWL-DL, each adding to the expressive power of the language. The TOWL layer cake is presented in Figure 1. In what follows, we present an overview of the different layers and their representational characteristics.

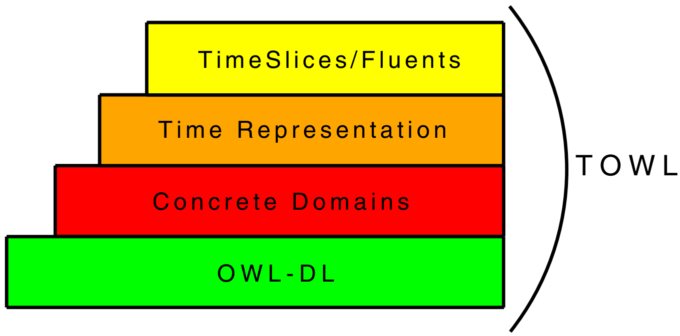



**Fig. 1.** The TOWL layer cake

| DL Notation | TOWL Abstract Syntax |
|---|---|
| $\exists u_1, u_2.p_d$ | dataSomeValuesFrom($u_1$ $u_2$ $p_d$) |
| $\forall u_1, u_2.p_d$ | dataAllValuesFrom($u_1$ $u_2$ $p_d$) |

**Fig. 2.** Existential and universal quantification in TOWL

The *concrete domains* layer enables functional role chains in the language, of the DL form shown below.

$$f_1 \circ f_2 \circ ... \circ f_n \circ g$$

Such chains consist of compositions of functional roles $f_i$ with a concrete feature $g$ that points to the concrete domain. Due to their nature, such chains are denoted as concrete feature chains in TOWL, and can be represented in TOWL abstract syntax as:

ConcreteFeatureChain($f_1$ $f_2$ ... $f_n$ $g$)

where $f_1, ..., f_n$ are abstract features and $g$ is a concrete feature pointing to some value in the concrete domain.

Based on such chains and concrete domain predicates, TOWL enables existential and universal quantification, as shown in Figure 2, where $u_i$ is a concrete feature chain and $p_d$ denotes a concrete domain predicate.

The *time representation* layer includes basic representations of time in the form of temporal intervals, as well as Allen's 13 interval relations that may hold between pairs of intervals. The core of this layer consists of a particular type of concrete domain in the form of a constraint system [10].

The *timeslices/fluents* layer, building upon the approach in [9], enables the representation of change and state transitions. This is achieved by extending the OWL-DL syntax and semantics to include timeslices and fluents. Additionally, the two other TOWL layers presented in this section contribute to the representational power of this layer by enabling more specific semantics at a concrete level. It thus becomes possible to express the fact that fluents may only connect timeslices that hold over the same interval [3].

Timeslices, as employed for the purpose of the TOWL language, are aimed at representing some concept (static individual) over a certain period of time (interval). Two main properties describe timeslices, namely the *timeSliceOf* property, that connects the timeslice to the static concept it represents, and the *time* property, indicating the time interval across which the timeslice holds. Additionally, fluent properties are employed to describe what is changing, i.e., what only holds true for a bounded period of time.

Fluents are the properties that hold for timeslices, and thus indicate what is changing. The semantics of fluents allows timeslices to be related to other timeslices, or to datatypes as attribute values. This differentiation in the range of fluents comes to address the proliferation of objects, inherent to the current approach. Hence, timeslices are created each time something is changing. However, creating timeslices for concrete values is deemed meaningless in the current context. Thus, for this case only, a timeslice may be connected to something else than another timeslice, namely a datatype. It should also be noted that the semantics of fluents enforce that the connected timeslices must invariably hold over the same time interval.

Finally, it should be remarked that the issue of cardinality has a strong relation with the identity of timeslices. For this purpose, two timeslices ($TS_1$ and $TS_2$) are defined as identical ($\mathtt{eq_{TS}}$), if the following holds true:

$$(TS_1, TS_2).\mathtt{eq_{TS}} \equiv (TS_1.\mathtt{time}, TS_2.\mathtt{time}).\mathtt{equal} \wedge \\ \wedge (TS_1.\mathtt{timeSliceOf}, TS_2.\mathtt{timeSliceOf}).\mathtt{sameAs}$$

## 4 Temporal Cardinality

The concept of temporal cardinality becomes relevant in the context of timeslices, as employed for the representation of change in the TOWL language. In what follows, we present an illustrative example of how the concept of cardinality changes semantics in a temporal context, and present a possible formalization of this concept.

The timeslices approach introduced in TOWL allows for the representation of change through the creation of timeslices. These temporal parts of individuals are related by fluents - the properties that connect timeslices and thus indicate what is changing and how. However, some of the OWL-DL constructs lose their meaning in this context of change, mostly due to the static semantics that pinpoint their definition. One such concept is the *cardinality* construct, present in OWL-DL in three closely related forms [11]:

- *minCardinality*: if stated to have the value $a$ on a property $P$, with respect to a class $C$, then any instance of $C$ will be related through $P$ to **at least** $a$ individuals (of which the type may further be restricted by the *range* of $P$);
- *maxCardinality*: if stated to have the value $a$ on a property $P$, with respect to a class $C$, then any instance of $C$ will be related through $P$ to **at most** $a$ individuals (of which the type may further be restricted by the *range* of $P$);

- *cardinality*: if stated to have the value $a$ on a property $P$, with respect to a class $C$, then any instance of $C$ will be related through $P$ to **exactly** $a$ individuals (of which the type may further be restricted by the *range* of $P$). In other words, both the *minCardinality* of $a$ and the *maxCardinality* of $a$ are simultaneously satisfied.

Moving to a temporal context, an extension of the static concept of cardinality may be envisioned in the sense that, at any point in time, only a restricted number of timeslices may describe a concept. In other words, temporal cardinality is meant to restrict the number of timeslices that may overlap, at any point in time for the same individual. These restrictions should be stated on fluents, with respect to static individuals whose timeslices are described by those fluents.

One example of a context where such restrictions are meaningful comes from the financial domain, and is also discussed in [9]. The example is built around the *Company*, *Person*, and *hasCEO* entities. In the current context we seek to represent the fact that, at any point in time, a company must have exactly 1 Chief Executive Officer (CEO), in the form of a person. This restriction targets two situations:

- *fluent cardinality*: the (static) cardinality of the *hasCEO* fluent should be equal to one, following the description above. In other words, the *hasCEO* fluent must be associated to exactly one timeslice of a static individual of type *Person* each time it is defined for a timeslice of an individual of type *Company*. This issue can easily be addressed by employing the OWL-DL *cardinality* construct, as done in [9];
- *overlapping timeslices*: the (temporal) cardinality of the *hasCEO* fluent should be equal to 1. In other words, at any point in time, the *hasCEO* relation must be described by one timeslice of a static individual of type *Person* [3].

These two situations are graphically illustrated in Figure 3, for the CEO example discussed above. Here *Company1-TS1* and *Company1-TS2* are timeslices of the static *Company1* individual, and *Person1-TS1* and *Person2-TS1* are timeslices of the *Person1* and *Person2* static individuals, respectively.

Extending the static concept of cardinality to a temporal setting, we introduce the following constructs:

- *temporalMinCardinality*: the equivalent of the *minCardinality* OWL-DL construct in a temporal setting;
- *temporalMaxCardinality*: the equivalent of the *maxCardinality* OWL-DL construct in a temporal setting;
- *temporalCardinality*: the equivalent of the *cardinality* OWL-DL construct in a temporal setting.

[3] It should be noted that the TOWL semantics enforce equal intervals for timeslices connected by a fluent.

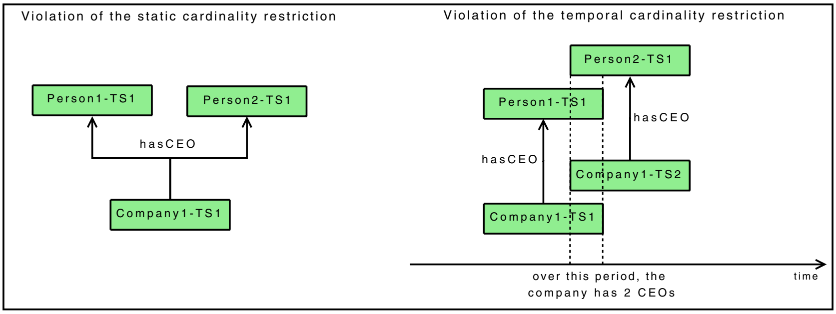



**Fig. 3.** Static vs. Temporal Cardinality

A more formal description for each of the introduced constructs is provided in definitions 1 through 3.

**Definition 1 (temporalMinCardinality)**
Given a fluent property $f$, a class $C$, an individual $i$ of type $C$ and a value $a$ such that $a \in \mathbb{N}$, we represent by $temporalMinCardinality(f, a)$ the restriction on $f$ with respect to timeslices of $i$ for which $f$ is defined that, at any point in time, any timeslice of $i$ is described by **at least** $a$ timeslices through $f$.

**Definition 2 (temporalMaxCardinality)**
Given a fluent property $f$, a class $C$, an individual $i$ of type $C$ and a value $a$ such that $a \in \mathbb{N}$, we represent by $temporalMaxCardinality(f, a)$ the restriction on $f$ with respect to timeslices of $i$ for which $f$ is defined that, at any point in time, any timeslice of $i$ is described by **at most** $a$ timeslices through $f$.

**Definition 3 (temporalCardinality)**
Given a fluent property $f$, a class $C$, an individual $i$ of type $C$ and a value $a$ such that $a \in \mathbb{N}$, we represent by $temporalCardinality(f, a)$ the restriction on $f$ with respect to timeslices of $i$ for which $f$ is defined that, at any point in time, $temporalMinCardinality(f, a)$ and $temporalMaxCardinality(f, a)$ simultaneously hold.

We next focus on giving a more formal semantic representation of the three types of temporal cardinality we introduced. In achieving this, we first define a function $g$ that, given a fluent $f$, a static individual $i$ and a point in time $t$, returns the number of timeslices of different individuals $j$ holding at $t$, for which $f$ is explicitly defined and linked from a timeslice of $i$ that also holds at $t$. The result of this function is a natural number, obtained by counting the unique individuals returned by the $g_{(f,i,t)}$ function.

$$\begin{aligned} g_{(f,i,t)} = |\{j \in C^{\mathcal{I}} \mid \exists x, y, s, e \texttt{ s.t. } x, y \in TS^{\mathcal{I}} \wedge (x, i) \in \texttt{timeSliceOf}^{\mathcal{I}} \wedge \\ \wedge\ (y, j) \in \texttt{timeSliceOf}^{\mathcal{I}} \wedge (x, y) \in f^{\mathcal{I}} \wedge s = \texttt{start}(\texttt{time}(y)) \wedge \\ \wedge\ e = \texttt{end}(\texttt{time}(y)) \wedge s \leq t \leq e\}| \end{aligned}$$

Moving to Description Logics, the semantics of the three constructs relating to temporal cardinality can be represented as follows, where $\geq_{\mathcal{T}}$, $\leq_{\mathcal{T}}$ and $=_{\mathcal{T}}$ denote *temporalMinCardinality*, *temporalMaxCardinality* and *temporalCardinality*, respectively, $a$, $f$ and $t$ preserve their meaning as previously, and $C$ denotes a concept.

$$(\geq_{\mathcal{T}}\ a\ f)^{\mathcal{I}} = \{x \in TS^{\mathcal{I}} \mid \forall t\ \exists i,\ i \in C^{\mathcal{I}} \wedge (x, i) \in \texttt{timeSliceOf}^{\mathcal{I}} \wedge g_{(f,i,t)} \geq a\}$$
$$(\geq_{\mathcal{T}}\ a\ f)^{\mathcal{I}} = \{x \in TS^{\mathcal{I}} \mid \forall t\ \exists i,\ i \in C^{\mathcal{I}} \wedge (x, i) \in \texttt{timeSliceOf}^{\mathcal{I}} \wedge g_{(f,i,t)} \leq a\}$$
$$(=_{\mathcal{T}}\ a\ f)^{\mathcal{I}} = (\geq_{\mathcal{T}}\ a\ f)^{\mathcal{I}} \cap (\leq_{\mathcal{T}}\ a\ f)^{\mathcal{I}}$$

Based on this definitions, a syntactic and semantic extension for TOWL can be proposed, that enable the representation of temporal cardinality when change and state transitions are represented by employing TOWL timeslices and fluents, as in the example presented in Figure 3.

## 5 Conclusions and Future Work

This paper provides a discussion of temporal cardinality in the context of the TOWL language. One of the challenges of representing such a concept is related to how the language enables the representation of change, i.e., by means of timeslices. This representation alone poses difficulties due to the fact that, unlike in the case of static cardinality, one should consider the case of overlapping timeslices in addition to just the property (or in this case fluent) alone. However, the representational power provided by the TOWL layers proves to be sufficient for enabling such a construct. Feature chains, as enabled by the *concrete domains* TOWL layer, and Allen's relations, as enabled by the *time representation* TOWL layer, allow the introduction of a function that enables counting the number of overlapping timeslices at some point in time, given a fluent and a static concept. Based hereon, we have shown that the static concept of OWL-DL cardinality can be extended with a temporal dimension, and more particularly in the context of the TOWL language.

Currently, the specification of the TOWL language is being finalized. Part of the future work will be aimed at building implementations for querying and reasoning support for TOWL. Furthermore, we aim at incorporating TOWL in a number of practical Semantic Web applications, such as StockBroker and CHI, as presented in the introductory part of this paper.

## Acknowledgement

The first author is supported by the EU funded IST STREP Project FP6 - 26896: Time-determined ontology-based information system for realtime stock market analysis. More information is available on the official website[4] of the project.

[4] http://www.towl.org

## References


1. Patel-Schneider, P., Hayes, P., Horrocks, I.: Web ontology language (OWL) abstract syntax and semantics (2004), http://www.w3.org/TR/owl-semantics/
2. van der Sluijs, K., Houben, G.J.: Relating user tags to ontological information. In: Proceedings of the 5th International Workshop on Ubiquitous User Modeling (UbiqUM 2008), Workshop at the 2008 International Conference on Intelligent User Interfaces (IUI 2008) (2008)
3. Milea, V., Frasincar, F., Kaymak, U.: Knowledge engineering in a temporal semantic web context. In: The Eighth International Conference on Web Engineering (ICWE 2008). IEEE Computer Society Press, Los Alamitos (2008)
4. Roddick, J.F., Patrick, J.D.: Temporal semantics in information systems - a survey. Information Systems 17(3), 249–267 (1992)
5. Pugliese, A., Udrea, O., Subrahmanian, V.S.: Scaling RDF with time. In: WWW 2008: Proceeding of the 17th international conference on World Wide Web, pp. 605–614. ACM, New York (2008)
6. Hobbs, J., Pan, F.: An ontology of time for the semantic web. ACM Transactions on Asian Language Information Processing 3(1), 66–85 (2004)
7. Pan, F., Hobbs, J.R.: Temporal aggregates in OWL-Time. In: Russell, I., Markov, Z. (eds.) FLAIRS Conference, pp. 560–565. AAAI Press, Menlo Park (2005)
8. Sotnykova, A., Cullot, N., Vangenot, C.: Spatio-temporal schema integration with validation: A practical approach. In: Meersman, R., Tari, Z., Herrero, P., Méndez, G., Cavedon, L., Martin, D., Hinze, A., Buchanan, G., Pérez, M.S., Robles, V., Humble, J., Albani, A., Dietz, J.L.G., Panetto, H., Scannapieco, M., Halpin, T.A., Spyns, P., Zaha, J.M., Zimányi, E., Stefanakis, E., Dillon, T.S., Feng, L., Jarrar, M., Lehmann, J., de Moor, A., Duval, E., Aroyo, L. (eds.) OTM-WS 2005. LNCS, vol. 3762, pp. 1027–1036. Springer, Heidelberg (2005)
9. Welty, C., Fikes, R.: A reusable ontology for fluents in OWL. In: Formal Ontology in Information Systems: Proceedings of the Fourth International Conference (FOIS 2006), pp. 226–336. IOS Press, Amsterdam (2006)
10. Lutz, C., Milicic, M.: A tableau algorithm for description logics with concrete domains and general tboxes. Journal of Automated Reasoning 38(1–3), 227–259 (2007)
11. Smith, M., Welty, C., McGuinness, D.: OWL web ontology language guide (2004), http://www.w3.org/TR/owl-guide/

# Author Index